2005

Integration of VMC++ into a Commercial Treatment Planning System

Joseph Kingsley Gardner
Virginia Commonwealth University

Integration of VMC++ into a Commercial Treatment Planning System

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science at Virginia Commonwealth University

by

Joseph Kingsley Gardner

B.S. Physics, University of Richmond, May 2001

Director: Jeffrey V. Siebers Ph.D.,

Associate Professor, Department of Radiation Oncology

Virginia Commonwealth University

Richmond, Virginia

December, 2005

Acknowledgments

I would like to thank Dr. Jeffrey Siebers for introducing me to this topic and for his constant encouragement along the way. I would like to thank my wife Angeline for her patience with my graduate school endeavors. I would like to thank my Aunt Jo for her strength and smile.

This research was supported in part by Varian Medical Systems and the National Institutes of Health, grant number 1R01CA98524

Table of Contents

List of Tables

Table of Figures

Abstract

Integration of VMC++ into a Commercial Treatment Planning System

By Joseph Kingsley Gardner M.S.

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science at Virginia Commonwealth University

Virginia Commonwealth University, 2005

Major Director: Jeffrey V. Siebers, Ph.D.
Associate Professor, Department of Radiation Oncology

Recently, there has been interest to integrate VMC++ into the commercial treatment planning system at VCU as another Monte Carlo code option, since it has been shown to increase efficiency dramatically without introducing a significant amount of systematic error. Also, independent validation of VMC++ for photon beams is of interest since this has not been performed previously in literature. This study included several tests required to integrate VMC++. Output factor normalization was performed and found to agree with experiment to within 1% for all field sizes except 1x1 cm^2. Geometric validation was successful. Dosimetric validation was performed with respect to DOSXYZnrc on a water phantom, resulting in agreement within statistical uncertainty except for slight differences at the surface of the phantom. Dosimetric comparison was made for a head-and-neck patient case, showing that 5% of the voxels did not agree within 2.8% of maximum dose. The ability of VMC++ to compute dose-to-water was compared to an in-house algorithm and found to agree within statistical uncertainty.

1 Introduction

1.1 Radiation Therapy Background

Radiation therapy first became possible when Wilhelm Roentgen discovered X-rays in 1895. At first, only low-energy radiation was available through use of specially designed vacuum tubes and expensively refined radium. While many positive outcomes were heralded, such as treatment of skin cancers, some unexpected side effects were also produced, such as skin reddening and scarring. Not until 1948 were megavoltage beams introduced by Donald Kerst's betatron. High energy X-rays proved to be a successful non-invasive modality against cancer. Not only did an external beam of these rays destroy tumors, but it also spared the skin from damage if used correctly.[1, 2, 3]

In designing an effective radiation treatment plan for a cancer patient, it is of utmost importance to understand where and how the dose will be deposited by the radiation. Planning of this treatment became much more precise with the advent of computerized tomography (CT) in the 1970's. By using CT, important volumes or structures in the body could be more precisely located and labeled. In an ideal treatment the tumor volume should be covered completely by a lethal dose, while other sensitive volumes such as the spinal cord should be limited to a low level of dose to avoid any damage or side-effects.

1.2 Treatment Planning

Treatment planning, simply put, is the planning of radiation treatment for a patient. Beam angles and the number of monitor units (a measure of machine output for a linear accelerator) are set such that the tumor volume(s) will receive a prescribed dose, while organs at risk will receive minimal dose. To aid in the planning, a treatment planning system (TPS) is used. The TPS is a computer program which incorporates treatment plan

information, including beam information, prescribed dose, and computed dose, along with patient CT scans.

Treatment planning has become much more complex recently with the advent of intensity-modulated radiation therapy (IMRT). In this method each incident beam is modulated such that some areas of the beam have greater intensity and some areas have lesser intensity. This modulation results in greater sparing of normal tissue while higher dosing of the tumor volume. In areas where the tumor is concentrated, the beam is focused more intensely, whereas in areas where the tumor is not as concentrated, the beam is focused less intensely.

1.3 Dose Calculation

When a treatment plan is prescribed, the dose must be calculated before delivered. Two important factors in this calculation are accuracy and time. There is no reason to use an extremely accurate dose calculation algorithm if it requires weeks to complete. On the other hand, if the process takes five minutes to complete, but is not accurate, error is introduced into the plan due to the lack of accuracy. Thus, it is of importance that the calculation be accurate, while not time consuming. While the overall accuracy of delivered dose has been recommended to be less than 5%[4], the accuracy of calculated dose has been recommended to be 3-4%.[5]

One way to gain insight into the distribution of dose from an external beam is to use a water phantom. Dosimeters may be placed strategically in the phantom to give an understanding of the distribution of dose. Since the density and average atomic weight of water closely resemble that of the human tissue, the distribution of dose in a water phantom may be used to predict how dose would be deposited in the human body. In order to understand how different materials in the body such as bone would affect the distribution, bone-like objects may be placed in the phantom as well. For clinical purposes, however, water phantom models are not practical considering the significant differences between patients. Not only is it impractical to insert dosimeters into a patient

for dose monitoring, but the resulting dose distribution also would not be accurate with a limited number of dosimeters.

Analytical dose calculation methods such as pencil beam, collapsed cone convolution, and superposition/convolution are current popular choices of dose calculation in TPSs. While these methods do not exactly model the way that dose is distributed in a plan, they give a reasonable approximation. Calculation time is less than an hour. Several factors are added into the calculation to account for lateral scatter and attenuation. However, it should be noted that limitations of these analytical algorithms exist in geometries containing complex heterogeneities.[6] At Virginia Commonwealth University (VCU) a superposition/convolution technique is used which only requires a few minutes to compute.

The dose calculation method with potentially the highest accuracy is Monte Carlo.[6, 7] A computer can simulate particles in a beam and predict how they will interact in a material. Since cross-sections and angular distributions of interactions have already been determined experimentally, the computer can use these data to correctly transport particles through media. These simulations have been called Monte Carlo simulations since the computer is using random selection to determine how the particles are transported. The Monte Carlo method will be more descriptively introduced in chapter two of this work. The two families of codes tested in this thesis, Electron Gamma Shower (EGS) and Voxel Monte Carlo (VMC) will be discussed more in depth in chapter two as well.

1.4 Thesis Objectives

VCU currently has a TPS capable of calling several Monte Carlo codes, including DOSXYZnrc of the EGSnrc family. Recently, there has been interest to integrate VMC++ into the system as another Monte Carlo code option, since it has been shown to increase efficiency dramatically without introducing a significant amount of systematic error. Also, independent validation of VMC++ for photon beams is of interest since this

has not been performed previously in literature. This thesis will cover the following tests necessary for integration of VMC++: geometric validation, phantom dosimetric validation, and patient dosimetric comparison with DOSXYZnrc. The accuracy and efficiency of VMC++ will be benchmarked to that of DOSXYZnrc, including a further study looking into the dose-to-water conversion used by VMC++.

Chapter two of this thesis gives a description of the Monte Carlo process, while also describing the history and specifics of VMC++. Chapter three explains how DOSXYZnrc and VMC++ are integrated into the TPS at VCU. In chapter four phase space source integration and output factor normalization are discussed. Phantom geometric validation is carried out in chapter five. Chapter six covers phantom dosimetric validation and patient dosimetric comparison. Chapter seven compares the dose-to-water calculation performed by VMC++ to that performed by an in-house VCU program. Conclusions are summarized in chapter eight.

2 Monte Carlo

2.1 The Monte Carlo Method

The Monte Carlo method involves transporting particles through a defined geometry. Each original particle that is transported through the geometry is termed a 'history'. When a particle is transported, the computer randomly decides several factors, including the distance to the particle's next collision, what type of collision it will undergo, and what angle and energy with which it will exit the collision. Each random choice the computer makes is called a 'sample'. A random number is chosen on the range [0,1], to which a probability density function (PDF) is mapped. An example of a collision-type PDF is shown in Figure 1.

The basic flow of a Monte Carlo simulation for each history is as follows: [8]

1) Sample the particle's initial energy, position, and direction from a source PDF.
2) Sample the distance to the particle's next collision, based on the material(s) through which the particle is being transported. Take note of the voxel (three-dimensional pixel) in which this collision occurs. If the distance places the particle outside of the geometry, terminate the particle history.
3) Sample the collision from the PDF corresponding to the surrounding material and the particle energy. If the sampled collision results in the particle being absorbed, terminate the particle history. If the particle is not absorbed, sample its outgoing direction and energy. If the particle's outgoing energy is below a cutoff energy, terminate the particle history. If the sampled collision produces a new particle, store the new particle on a stack for its own sampling later.
4) Score a correct deposition of dose in the voxel in which the particle interacted.
5) Repeat steps two through four until the particle is absorbed or leaves the geometry.

6) For every extra particle created by a collision in step three, cycle through steps two through five for the new particle until its history is terminated.
7) Add together all depositions of dose for particles existing in the current history.
8) Calculate the sum of all history scores, along with respective absolute uncertainties, defined as

$$s_{\bar{x}} = \sqrt{\frac{1}{N-1}\left(\frac{\sum_{i=1}^{N} X_i^{\,2}}{N} - \left(\frac{\sum_{i=1}^{N} X_i}{N}\right)^2\right)}$$

where N is the number of primary histories and X_i is contribution to the scored dose by independent history i.[9]
9) Repeat steps one through eight for a total number of histories or until a desired uncertainty is obtained.

To understand the finer details of Monte Carlo sampling, the reader is encouraged to read Williamson's chapter in *Monte Carlo Simulation in the Radiological Sciences.*[10]

A major roadblock to Monte Carlo simulation lies in the amount of time required for its execution. One source of spent time lies in all of the sampling required per particle. Each particle must have a distance, direction, and collision sampled for every collision until it is absorbed. Since photons only undergo several interactions before being absorbed, the time spent sampling for them is not critical. However, for electrons, which typically undergo ~10^6 collisions before absorption, the sampling time simply requires too much computing time.

Variance reduction techniques (VRTs) are used to reduce the amount of time required for a Monte Carlo simulation, while not sacrificing the accuracy of the calculation. VRTs are implemented in order to gain efficiency, which is defined as

$$\varepsilon = \frac{1}{\sigma^2 t}$$

where σ^2 is the statistical variance and t is the computing time required for the simulation. The most common VRT used in Monte Carlo simulations is electron

condensed history (CH). This technique, first proposed by Berger[11] in 1963, drastically reduces the amount of time required to transport electrons by realizing that the majority of the collisions that electrons experience only slightly affect their energy and direction. Samples are only taken at every major "knock-on" collision, thereby increasing efficiency, while not introducing significant systematic bias. It was not until two decades later that CH was theoretically justified by Larsen[12]. A second common VRT is found in history repetition. Using this technique, a history may be repeated several times as long as it is started at a different location each instance. Every time the history is repeated, efficiency is increased since no new sampling needs to be performed.[13]

Even though Monte Carlo's accuracy is potentially higher than that of conventional algorithms, its accuracy comes at the cost of computation time. Since its statistical uncertainty is indirectly related to the number of histories squared, computation time must be quadrupled in order to reduce the uncertainty by a factor of two. However, as advances in computer technology are made, and as more VRTs are utilized, the computation time decreases accordingly. In order to replace analytical dose calculation methods in the clinic, the computation time of Monte Carlo must be lowered to a level comparable to that of the analytical methods.

2.2 EGSnrc/DOSXYZnrc

One of the most commonly used Monte Carlo codes is the Electron-Gamma-Shower (EGS) family of codes.[14] In the early 1970's the first version of EGS was produced by Ford and Nelson. It was then benchmarked against various experiments and compared to other Monte Carlo results, and found to have good agreement. Since then, the code has been improved several times (EGS2 in 1975, EGS3[15] in 1978, EGS4[16] in 1985, EGSnrc[17,18] in 2000), while being benchmarked continuously. For this reason, it is considered a good standard of comparison for other Monte Carlo transport codes.

DOSXYZnrc[19] is a Monte Carlo simulation code, which uses voxels to calculate a dose distribution. It is a member of the EGSnrc family of codes and was upgraded from

DOSXYZ in 2001. DOSXYZnrc is a very flexible code insofar that it can have several beams incident on the phantom/patient while also being able to use phase space input files.

2.3 Voxel Monte Carlo

In the mid 1990s Kawrakow, Fippel, and Friedrich[20] developed a new electron transport Monte Carlo code, which they named Voxel Monte Carlo (VMC). Their code was specifically for use of radiation therapy dose calculations in voxelized phantoms. One of the main reasons they did this was because they realized the conventional code being used at the time, EGS4, was designed to describe electron transport in all materials in a very wide energy range (from a few keV to GeV energies). They reasoned that for radiation treatment planning (RTP), only materials of density less than $3 g/cm^3$ and energies in the 1-30 MeV range were of interest. Using several simplifications and approximations, a 35x higher efficiency was obtained for electron transport compared to EGS4 for electron beam simulations. First of all, they introduced history repetition as a VRT, while only adding less than one percent of systematic error. Secondly, they reduced the average number of electron steps per history by allowing electrons to proceed through boundaries in a single CH step. This is a two step process. First, electrons are transported through a homogeneous water medium with different CH steps defined in terms of energy loss. Secondly, these histories are applied to a voxelized heterogeneous geometry. This application requires rescaling of the path length of each step by a ratio of the voxel-dependent stopping power to that of water, thereby keeping the energy lost in the step equal to that lost in the step in water. The energy deposited in each voxel is calculated by multiplying the stopping power of the voxel by the distance traversed by the electron through the voxel. In addition to the energy loss, the electron scattering angle at the end of the step requires rescaling due to the material dependence of the scattering angle. Utilizing these methods, several voxels of various materials can be crossed in one CH step. The reduction of total CH steps per history resulted in an

efficiency gain of up to 150 times[21] over EGS4, which terminates CH steps at every interface using its boundary crossing algorithm, PRESTA[22] (parameter-reduced electron step transport algorithm). Thirdly, bremsstrahlung photons were discarded since their contribution to the overall dose was insignificant in RTP range. Whenever a bremsstrahlung photon was produced, its energy was simply subtracted from the electron that produced it. Fourthly, the previously complex cross section for producing delta particles was simplified while producing negligible effects. Finally, it was noted that even with an underestimation of large-angle elastic scattering due to the previous simplification, the dose distribution was unaffected. Another improvement of VMC was that it required no large amounts of precalculated data. Because of VMC's dramatic increase in efficiency and subsequent decrease in computing time, its creators boldly concluded: "We therefore consider the VMC model to be a serious alternative to pencil-beam algorithms."

In 1998 Fippel[23] went on to expand the horizons of VMC by applying it to photon transport as well. Because of the inclusion of photon transport, he renamed the code XVMC. When compared to EGS4, XVMC was more efficient by a factor of 15-20. This increase in efficiency was due to: 1) the previous increase in efficiency for electron transport in VMC, 2) a fast ray tracing technique, and 3) the initial ray tracing technique, which determined the number of electrons created in each voxel by the primary photon beam. In 1999 Kawrakow and Fippel[24] applied several more VRTs to improve XVMC: 1) interaction forcing: photons were forced to interact before they exited the geometry. Otherwise, all of the time sampling for the photon would be wasted. 2) Quasi-random sequences were used instead of pseudo-random numbers. These sequences are more efficient at uniformly filling a space. 3) Woodcock tracing[25] was used so that heterogeneous media would not require time-consuming ray tracing. In this method, the photon is transported through a heterogeneous medium as though it were a homogeneous medium formed by the most dense material from the heterogeneous medium. The photon is then weighted by the ratio of the attenuation coefficient of the material in which the photon is initially located divided by that of the most dense material. 4) Photon splitting,

similar to history repetition for electrons, was used. An overall efficiency gain of 3.5-11 was made over the previous XVMC code. The new XVMC code had an efficiency 50 to 80 times higher than EGS4.

Kawrakow[26] introduced the newest installation of VMC in 2000: VMC++. This new version was written using C++ and benefited from its object-oriented design. Every interaction relevant to the simulation was represented by a separate interaction object. Implementation of improvements and modifications to the code also became much easier. VMC++ became a much more flexible code than its predecessors. The major new VRT introduced in VMC++ was the use of Simultaneous Transport Of Particle Sets (STOPS). This technique was similar to history repetition, but instead of just repeating one electron at a time, entire sets of electrons with equal energies were repeated. By using STOPS, several variables such as interpolation indexes, azimuthal scattering angles, and distances to the next collision could be sampled just once for an entire set of particles. The largest gain in efficiency was achieved when roughly forty electrons per set were transported by STOPS. VMC++ also extended the use of quasi-random sequences to electron beams. When combined with STOPS, the quasi-random sequences yielded an efficiency gain of about five times.

3 MCV Monte Carlo Flow

The TPS used at VCU is ADAC Pinnacle (Philips Medical System). Either a user-defined phantom may be created or CT slices may be imported for dose calculation. Multiple beams can be simulated at arbitrary source-to-surface distances (SSDs) and at user-defined gantry, collimator, and couch angles rotated about a user-defined isocenter (the point about which the gantry rotates). The prescription can be set at a certain amount of monitor units (MUs). The dose grid may also be created, thereby defining voxel size. After simulation, user-defined isodose lines may be displayed over top of the phantom or CT image. One useful feature of Pinnacle is its ability to call other codes. One of its side menus shows a list of user-defined 'hot scripts', which serve a variety of functions. At VCU, hot scripts have been developed which range from extracting depth dose data to calling Monte Carlo dose calculation codes. Pinnacle also allows for planning of multiple trials, each with its own specific treatment plan.

The system of programs used to perform various Monte Carlo codes at VCU was developed by Siebers *et al.*[27] This system is referred to as mcvmc (Medical College of Virginia Monte Carlo). A flow diagram is shown in Figure 2. Mcvmc is a program written in the C++ programming language which reads beam parameters from Pinnacle, creates input files for Monte Carlo codes which simulate the treatment delivery, submits jobs to (multiple) selected computers to perform the Monte Carlo simulation, collects and merges results from the MC simulations, and inputs the final dose distribution back into Pinnacle. By selecting different options in mcvmc, one can tailor the Monte Carlo codes which are run for a given test case (e.g. can select BEAM[28] or BEAMnrc, or can select DOSXYZ, DOSXYZnrc, or VMC++). In this study mcvmc was used to ensure that DOSXYZnrc and VMC++ would use the same incident particles on phantom/patient geometry, so that any differences in results could be attributed to the differences in those two codes.

3.1 Graphical User Interface for mcvmc

The 'hot script' which is used to call VMC++ is called mcvmc.vmc (mcvmc, with VMC++ capabilities). This script calls the mcvmc.vmc.tcl file, which produces a graphical user interface (GUI) between Pinnacle and mcvmc, which allows the user to set options and flags to be passed to mcvmc, as seen in Figure 3. The user may either run Monte Carlo with the standard defaults, or he/she may alter them in the Config menus to setup the desired simulation.

The general config window is seen in Figure 4. The first option, 'Compute Density Grid', sets a flag to run ct_create[19], a code that derives materials from CT information. At VCU the CT information is translated into a range of fifty-six materials, which very accurately model the various tissues found in the human body. The second option, 'Compute Phase Space', sets a flag to run BEAM or BEAMnrc, codes which simulate particle transport through the accelerator head and store the resulting particles in a phase space. The third option, 'Compute Patient Dose', sets a flag to run the Monte Carlo code, which transports the particles through the phantom/patient CT. The fourth option, 'Load Dose Files', sets a flag to load the dose back into Pinnacle so that the user may view the resulting dose distribution. The first pull down menu allows the user to choose which code to generate the phase space with: BEAM or BEAMnrc. In this study BEAM was used. The second pull down menu allows the user to choose which code to simulate particles through the phantom/CT images: DOSXYZ, DOSXYZnrc, MCVRTP, or VMC++. Two pull down menus further down allow the user to choose which computer to run mcvmc from, and which appropriate cluster to run the Monte Carlo subprocesses on. Finally, the user may define the desired relative uncertainty to run the Monte Carlo code to; the default value is set at two percent.

The more specific config menus, the DMLC (dynamic multi-leaf collimator) config menu, the VMC++ config menu, and the research config menu, are seen in Figure 5, Figure 6, and Figure 7, respectively. The DMLC config menu allows the user to

choose if and how particle transport through the DMLC should be simulated, as in the case when using IMRT fields. The VMC++ config menu lets the user choose whether VMC++ should compute dose-to-water or dose-to-material (see chapter 7 below). Also, the user may choose whether a source model is being used or if particles are being sampled from a phase space. The photon splitting factor (a type of VRT) may either be set to equal that of DOSXYZnrc or independently set by using the slider at the bottom of the menu. Finally, the research config menu gives the user the option of deleting old phase space data files, causing the subsequent mcvmc run to calculate new ones. If this option is left unchecked, the subsequent mcvmc run will simply use the old files. Also of importance is the option to run the beta version of the code, which is valuable for testing new changes to the code.

Another important function of the mcvmc.vmc.tcl code is to create a basic VMC++ input file (see Appendix A). This input file records several data for VMC++. First of all, it records whether a source model is being used to simulate particles or if particles are being taken from phase space (see below for an explanation of phase space). In this study, particles were always taken from phase space, which was previously generated by a BEAM simulation through the accelerator head. Secondly, the input file records beam geometry and secondary jaw positioning. Also of importance, it records the phantom CT data file, dose scoring options, variance reduction options, and the desired uncertainty.

A final important function of the mcvmc.vmc.tcl code is to call the next code in the chain, tcl_to_mcv (see next section). A new terminal window is opened, and all of the flags and values set by the user in each of the config menus as well as any default flags and values are passed to tcl_to_mcv.

3.2 Parallelization of the Process

Tcl_to_mcv is a c-shell code that is called by mcvmc.vmc.tcl and is passed all of the flags and values from that code. Since the simulation will not necessarily be run on the same

computer on which Pinnacle is being run, scripts containing all of the flags and values must be copied to the appropriate machine(s). The tcl_to_mcv code then calls either the mcvmc code or the mcvmc.beta code, depending on whether the beta version was selected or not, and passes along the flags and values.

3.3 mcvmc Code

The mcvmc code is a C++ code, which first determines if the flags and values passed to it are logically consistent with each other and if they are in the proper ranges. It then loops over all beams and calls a subroutine, compute_mcvmc_beams, to perform the selected processes. Upon successful completion, mcvmc reports normal termination. The core of the run lies in the compute_mcvmc_beams subroutine, where commands are given to run each beam through the accelerator head, and then through the phantom geometry using the simulation code previously selected by the user. This is also where the photon splitting factor is computed such that VMC++ replicates DOSXYZnrc, if the user desired so. Finally, the time elapsed for each run is displayed.

3.4 Calling VMC++

If the VMC++ flag is set, runVMC is called. This code substitutes the last several variables into the VMC++ input file created earlier by mcvmc.vmc.tcl. These variables include the phase space file name, the VMC++ phantom file name, the phase space z-distance, the number of particles to be run, the photon splitting factor (which was derived in compute_mcvmc_beams.cc), and two random seeds. After the input file is edited, runVMC calls runVMCStandAlone, which remotely executes VMC++ on the input file. Once VMC++ produces a dose distribution, another program is required to convert the VMC++ dose distribution to one that is readable by Pinnacle, since the two have different formats. It should be noted that all other available Monte Carlo codes are implemented in a similar fashion as VMC++ (i.e. runBEAM, runDOSXYZ, etc...).

4 Integration of VMC++

4.1 Phase Space Source Integration

Code was developed so that a phase space source could be integrated into VMC++ as an allowable input. A phase space is a file which contains information about particles, including position, direction, energy, and weight (how much each particle is weighted). The particles which are transported through the accelerator head by BEAM are stored in a phase space and subsequently used as a source of incident particles to be transported through the phantom geometry by a dose calculation code. The phase space source code was modified by Siebers so that it would fit into the mcvmc family of codes, as well as perform a rotation/transformation changing the accelerator coordinates to the patient coordinates before transport through the phantom was carried out. The rotation matrix used was borrowed from an in-house code which had required the exact same rotation for DOSXYZ.

4.2 Output Factor Normalization

Since Monte Carlo computes dose per incident particle, and not dose per MU, the output must be multiplied by a particles-per-MU factor. This factor must be scaled properly such that VMC++ results are consistent with DOSXYZnrc. If the VMC++ result is multiplied by the correct factor, all field size distributions should match DOSXYZnrc predictions.

4.2.1 Method

Pinnacle was used to create a 10x10 cm^2 6MV photon beam orthogonally incident on a rectangular water phantom (48cm x 48cm x 41cm depth), which resembled a Wellhofer

3d phantom (Scanditronix Wellhofer North America). Since the resolution of the depth dose curves were of primary importance, the dose grid was created to form voxels with a cross section of 2x2 cm^2, and depth of 0.5 cm. The voxels completely covered the depth of the phantom while also covering 46 cm of the 48 cm length and width. Both DOSXYZnrc and VMC++ were run using the same particles from phase space to a nominal statistical uncertainty of 0.5%. In order to calculate the particles-per-MU factor, a method proposed by Siebers *et al*[29] was used. The DOSXYZnrc and VMC++ depth dose curves were integrated from 5 cm to 15 cm depth. The VMC++ results were then multiplied by the ratio of the integrated DOSXYZnrc dose divided by the integrated VMC++ dose.

For the following field sizes, the VMC++ output was multiplied by the ratio determined for the 10x10 cm^2 field by the process above. Field sizes of 1x1 cm^2, 2x2 cm^2, 4x4 cm^2, 10x10 cm^2, 20x20 cm^2, and 40x40 cm^2 at energy 6MV were simulated, and field sizes of 1x1 cm^2, 10x10 cm^2, and 40x40 cm^2 at energy 18MV were simulated. It should be noted that for fields smaller than 10x10 cm^2, voxel sizes were reduced to 0.3 cm in the lateral directions to yield higher in-beam resolution.

Output factors normalized to the 10x10 cm^2 field were calculated for both DOSXYZnrc and VMC++. The DOSXYZnrc and VMC++ output factors were then compared to experimental data collected using a Wellhofer 3d water phantom and IC3 ionization chambers (from Wellhofer).

4.2.2 Results

It was found that the integral of the depth dose curve of VMC++ between 5 cm and 15 cm depth differed from that of DOSXYZnrc by a factor of $1.002 \times 10^{-10} \pm 0.004 \times 10^{-10}$ for the 10x10 cm^2, 6MV field. After communication with the author of VMC++, it was learned that VMC++ produces dose distributions multiplied by a factor of 1.0×10^{-10} Gy/incident particle. Since the results were within statistical uncertainty of this factor, 1.0×10^{-10} was used as the normalization factor in all other calculations. When this factor

was multiplied into all VMC++ calculations, the depth dose curves of the two Monte Carlo codes lined up within statistical uncertainty. The depth dose curves for the 1x1 cm^2, 10x10 cm^2, and 40x40 cm^2 fields at energy 6MV may be seen in Figure 8. The depth dose curves for the 2x2 cm^2, 4x4 cm^2, and 20x20 cm^2 fields at energy 6MV may be seen in Figure 9. The depth dose curves for the 1x1 cm^2, 10x10 cm^2, and 40x40 cm^2 fields at energy 18MV may be seen in Figure 10.

At several field sizes, VMC++ noticeably (i.e. greater than two standard deviations) underpredicts the surface dose calculated by DOSXYZnrc. This underprediction is due to the differing ways in which the codes handle electrons as they pass under the electron cutoff energy. For DOSXYZnrc, when an electron's energy passes below the cutoff energy, it is terminated and the dose is deposited in the current voxel. For VMC++, when the electron's energy passes below the cutoff energy, the electron takes a step equal to its residual range, which is assumed to be a straight line[24]. The dose is then deposited along this step. To test if the difference in handling electrons when they pass below the cutoff energy is the reason for the differing surface doses, DOSXYZnrc was run at three different cutoff energies (887 keV, 189 keV, and 50 keV kinetic energy), while VMC++ was run at two different cutoff energies (887 keV and 43.6 keV).. Figure 11 plots the calculated surface doses of the two codes as a function of the electron cutoff energy. The predicted surface doses converge as the electron cutoff energy is reduced, confirming that the differences in how the codes deal with electron transport at cutoff energies is the cause of the surface dose differences. For all other calculations in this thesis, DOSXYZnrc was run with an electron cutoff energy of 189 keV kinetic energy, the standard cutoff energy for the clinic at VCU. VMC++ was run with its own computed cutoff energy, usually in the vicinity of 880 keV for water phantoms or 300 keV for heterogeneous patient geometries.

Output factors computed by DOSXYZnrc and VMC++ were compared to those obtained with experimental data collected using IC3 ion chambers in a Wellhofer 3d water phantom. Figure 12 and Figure 13 show the results for the 6MV and 18MV beams respectively. The difference between the Monte Carlo-computed output factors and

experiment for the 1x1 cm^2 field at energy 6mv may be attributed to the uncertainty involved in positioning of the dosimeter in experiment. The difference might also be corrected for by more precisely modeling the IC3 detector in the Monte Carlo calculations.[30]

5 Phantom Geometric Validation

5.1 Introduction

In order to properly integrate a Monte Carlo code, geometric validation must be performed.[31] Since the accelerator beam head and the phantom have different coordinate systems (see Figure 14), the TPS must carry out coordinate transformations. Therefore, when the TPS simulates a beam, it must be verified that VMC++ reads in the beam at the proper location and at the proper angle in relation to the phantom, so that the dose will be displayed in the correct location.

When performing geometric validation, irregularity allows for easier verification. If one were to use a square symmetric field orthogonally incident on a cubic phantom, there could be several possible false positive outcomes: essentially two wrongs could possibly make a right due to degenerate results. Therefore, it is important to use an asymmetric field incident on an irregular phantom at an odd angle with the isocenter not located at the center of the phantom. If each side of the phantom is uniquely irregular, any degenerate results would be eliminated.

Several tests must be carried out for successful geometric validation. First of all, the gantry, collimator, and couch angles must each be tested separately at unique angles. Next, there should be a test in which all three angles are tested together at unique angles to ensure that there is no error introduced by coupling multiple rotations. Finally, a test must be carried out to test that the SSD is being set properly. A summary of these tests may be seen in Table 1.

5.2 Method

Pinnacle was used to simulate a 6MV photon beam with a field shape as seen in Figure 15. By hand-drawing the contour of the water phantom, unique irregularity was ensured for each side of the phantom, as well as for every slice through the phantom. The

isocenter was placed on the face of the phantom into which the beam was directed. The source-to-axis distance (SAD) was set to 100cm, resulting in a source-to-surface distance (SSD) of 100cm as well. Voxel sizes were set to 0.5 x 0.5 x 0.5 cm^3. For the gantry angle test, the gantry angle was set to 53^o, while all other angles were set to 0^o. For the collimator angle test, the collimator angle was set to 128^o, while all other angles were set to 0^o. For the couch angle test, the couch angle was set to 12^o, while all other angles were set to 0^o. For the multiple angle test, the gantry angle was set to 53^o, the collimator angle to 128^o, and the couch angle to 12^o. For the SSD test, the SSD was set to 60cm, while all angles were set to 0^o. VMC++ was run to a nominal statistical uncertainty of five percent in all tests.

5.3 Results

All transverse, sagittal, and coronal slices were scanned through visually, and the dose distribution was found to line up with the aperture outline indicated by Pinnacle. Also, the expected dose fall-off was observed in the air cavities scattered throughout the phantom. Sample transverse, sagittal, and coronal slices may be seen for the gantry angle test, collimator angle test, couch angle test, multiple angle test, and SSD test in Figure 16, Figure 17, Figure 18, Figure 19, and Figure 20, respectively.

6 Dosimetric Validation and Comparison

In order to verify that VMC++ predicts correct dose distributions, comparisons must be made either with experiment or with a Monte Carlo code which has been well benchmarked against experiment, in this case DOSXYZnrc. In order to compare the two dose distributions, four methods were used. First of all, dose profiles were created to provide a visual check. However, since dose profiles only offer a two-dimensional comparison, two other comparison tools, gamma indices and a Kawrakow-Fippel method, were also used. Finally, dose volume histograms (DVHs) were produced for patient cases.

6.1 Gamma Indices

Gamma indices were first introduced as a means of quantitative comparison of two dose distributions by Dan Low in 1998[32]. This method combines the dose difference method and the distance-to-agreement (DTA) method. The dose difference method simply produces a percent difference between the two dose distribution values on a voxel-by-voxel basis. The weakness of this method occurs in areas of steep dose gradient, where a slight lateral offset of the distributions results in a large percent dose difference. The DTA method accounts for this weakness by finding the closest voxel of agreement between the two distributions. Therefore the DTA method produces a voxel-by-voxel grid of distances. When encountering a steep dose gradient, while the dose difference method could result in a large difference, the DTA method could result in a small difference, on the order of millimeters. The weakness of the DTA method occurs, however, in areas of shallow dose gradients. In this case, whereas the dose difference method would produce a relatively small percent difference, the DTA method may produce a large distance-to-agreement between voxels. One possible way to combine the two methods is by creating a grid which shows where either one of the two methods

creates a passing score (i.e. 2%, 2mm). As long as a voxel has less than a 2% difference *or* less than a 2mm DTA, it will be given a passing value. If the voxel fails both tests, then it will receive a failing value. The problem with this combination of the two methods is that it is a binary system: it would be difficult to determine the magnitude of failure for each voxel. To compensate for this issue, Low[32] introduced the gamma index, a non-binary combination of the dose difference and DTA methods. Table 2 shows the derivation of the gamma index. Similar to the binary system mentioned above, dose difference and DTA criteria are set for the gamma analysis. A voxel has an acceptable tolerance if its gamma index is less than one. Code was written by Fix to implement gamma analysis at VCU.

Before conclusions may be drawn using gamma analysis between two dose distributions calculated by two different Monte Carlo codes, it is important to know what the gamma analysis yields when comparing two dose distributions calculated by a single Monte Carlo code. Pinnacle was used to simulate two identical 10x10 cm^2 6MV beams orthogonally incident on a water phantom. VMC++ was then used to calculate the absorbed dose in the phantom for each beam. A gamma analysis was performed on the resulting dose distributions. The results, seen in Table 3, show that for the 1% / 1mm criteria, only 99.3% of the voxels passed the gamma analysis. This percentage may be used as a reference when using gamma analysis to examine differences between distributions produced by different codes.

6.2 Kawrakow-Fippel Method

Kawrakow and Fippel introduced another means of quantitatively comparing dose distrubitions in 1999[24]. They defined x_{ijk}

$$x_{ijk} = \frac{D_{ijk}^{ref} - D_{ijk}^{eval}}{\Delta D_{ijk}}$$

as the dose difference between the reference and evaluated distributions in voxel *ijk* divided by the combined statistical uncertainty ΔD_{ijk} (summed by quadrature). If the two

distributions only differed randomly, then x_{ijk} would be fitted well by a Gaussian function centered at zero. In the case that systematic deviation existed between the two distributions, the moments of the distribution of x_{ijk} yielded information concerning the amplitude and frequency of voxels containing significant systematic deviation. Kawrakow and Fippel used a fitting function to interpret the x_{ijk} distribution. The probability distribution $f(x)$ to find a voxel with a deviation of x was given by

$$f(x)=\frac{1}{\sqrt{2\pi}}[\alpha_1\cdot e^{-\frac{(x-\delta_1)^2}{2}}+\alpha_2\cdot e^{-\frac{(x-\delta_2)^2}{2}}+(1-\alpha_1-\alpha_2)\cdot e^{-\frac{x^2}{2}}].$$

Simply put, a fraction α_1 of the voxels had a systematic deviation of δ_1, another fraction α_2 had a systematic deviation of δ_2, and the rest of the voxels had no systematic deviation. By fitting the probability distribution to x_{ijk}, a quantitative comparison may be made between two dose distributions.

6.3 Dose Volume Histograms

For patient geometries, another method to compare two dose distributions is found in dose volume histograms (DVHs). With this method, the dose received to important structures, such as the gross tumor volume or the spinal cord, is plotted on the x-axis against the volume of the structure receiving a level of dose on the y-axis. The volume may be normalized such that it displays the proportion of the structure that is receiving a level of dose. Also, the DVHs for two distributions may be plotted on the same graph, making for a good visual comparison.

6.4 Phantom Case

Pinnacle was used to create a 10x10 cm^2 6MV photon beam orthogonally incident on a rectangular water phantom (48cm x 48cm x 41cm depth). Voxel sizes were set to 0.5 x 0.5 x 0.5 cm^3 and completely covered the depth of the phantom while also covering 46 cm of the 48 cm length and width. Both the DOSXYZnrc and the VMC++ codes were

run to one-half percent nominal statistical uncertainty using the mcvmc family of codes discussed in section two. The VMC++ photon-splitting factor was set equal to that used by DOSXYZnrc. The 'Clean Up' flag was left unchecked, allowing for re-use of the exact same particles from phase space in both codes.

Depth dose profiles were generated along the beam axis and may be seen above (Figure 8, Figure 9, and Figure 10). Also, lateral profiles were produced at depths of 2cm, 5cm, and 10cm for the 10x10 cm^2 6MV beam. The lateral profiles in the x-direction of the patient coordinate system may be seen in Figure 21, while the lateral profiles in the z-direction of the patient coordinate system may be seen in Figure 22. The DOSXYZnrc and VMC++ dose profiles agreed well with each other, with the exception of the difference in surface dose mentioned earlier.

The gamma index results are seen in Table 4 for voxels with dose greater than fifty percent of the maximum dose (recommended by Rogers and Mohan[33]). Whereas 100% of the voxels passed the 2% / 2 mm criteria, 98.6% passed the 1% / 1mm criteria. A sagittal slice of the gamma distribution revealing the surface dose difference may be seen in Figure 23. MATLAB was used to fit the data for the Kawrakow-Fippel method results. The fit may be seen in Figure 24. According to the fit, a fraction $\alpha_1 = 0.4$ of the voxels have a systematic difference of $\delta_1 = 0.27$ standard deviations. The value for α_2 corresponded to zero. The remainder of the voxels showed no systematic deviation. Since the standard deviation in this case referred to a combined nominal statistical uncertainty of 0.7%, the difference corresponds to a 0.2% difference of the maximum dose, which is insignificant.

The total CPU time required to run the DOSXYZnrc calculation was just over one day. The total CPU time required to run the VMC++ calculation was about forty-five minutes. Accounting for subtle differences in respective variances, VMC++ yielded an efficiency approximately 37 times greater than that of DOSXYZnrc. It should be noted that a greater gain in efficiency would be possible if the photon-splitting factor of VMC++ were set to an optimal level instead of using the value set by DOSXYZnrc.

6.5 Patient Case

A comparison between DOSXYZnrc and VMC++ was also carried out in a patient case to observe how VMC++ performed in a more complex situation (i.e. with several IMRT fields incident on a heterogeneous phantom). A head-and-neck (H/N) patient plan was loaded into Pinnacle for this comparison. The plan incorporated nine IMRT beams located at gantry angles of 0^{o}, 30^{o}, 125^{o}, 165^{o}, 190^{o}, 235^{o}, 270^{o}, 300^{o} and 330^{o}. The dose grid had voxel sizes of 0.4 x 0.4 x 0.4 cm^3, the standard voxel size for patient dose computations at VCU. The same particles from phase space were used in both the DOSXYZnrc case and the VMC++ case. However, the transport of these particles through the DMLC effectively differentiated them before they interacted with the patient geometry, since different random seeds were generated for each simulation of transport through the DMLC.

The gamma index results comparing the DOSXYZnrc and VMC++ total dose distributions are seen in Table 5, as well as a sagittal slice of the distribution in Figure 25. Whereas 99.1% of the voxels passed the 2% / 2 mm criteria, only 86.4% passed the 1% / 1mm criteria. The only areas in which the gamma index was greater than one were in air cavities within the body or in the air directly in front of the surface of the patient.

MATLAB was used to fit the data for the Kawrakow-Fippel method results. The fit may be seen in Figure 26. According to the fit, a fraction $\alpha_1 = 0.0528$ of the voxels had a systematic difference of $\delta_1 = 4.06$ standard deviations. The value for α_2 corresponded to zero. The remainder of the voxels showed no systematic deviation. Since the standard deviation in this case referred to a combined statistical uncertainty of 0.7% of maximum dose of the two calculations, the difference corresponds to a 2.8% difference of the maximum dose. When combined with the results of the gamma index display, it is seen that these voxels are located mostly in air cavities within the patient. The DVHs comparing the two distributions may be seen in Figure 27. VMC++ noticeably underpredicted DOSXYZnrc. The dose to 90% (D_{90}) of the clinical target volumes (CTVs) agrees within one percent.

The total CPU time required to run the DOSXYZnrc calculation was just over two and a half days. The total CPU time required to run the VMC++ calculation was about two hours. Accounting for subtle differences in respective variances, VMC++ yielded an efficiency approximately 35 times greater than that of DOSXYZnrc.

7 Dose-to-Water Analysis

Historically, dose measurements have been computed in terms of dose-to-water (D_{water}). Even though the human body is comprised of a variety of tissues, clinically the absorbed D_{water} is used. However, most Monte Carlo codes calculate dose-to-material ($D_{material}$) instead of D_{water}, since they are transporting through patient materials. There is debate as to whether continuing to use D_{water} or starting to use $D_{material}$ should be done. In any case, an algorithm is required to convert $D_{material}$ to D_{water} in scenarios when dose distribution comparisons are desired.[34, 35]

One of VMC++'s capabilities is to directly convert $D_{material}$ to D_{water} during execution of the transport. VMC++'s computation of D_{water} was compared to an in-house algorithm developed by Siebers *et al.*[34] According to Bragg-Gray cavity theory,

$$D_{water} = D_{material} \left(\frac{\overline{S}}{\rho}\right)^{water}_{material}$$

where $\left(\frac{\overline{S}}{\rho}\right)^{water}_{material}$ is the unrestricted water-to-material mass collision stopping power averaged over the energy spectra of electrons at the point of interest. Siebers found that $\left(\frac{\overline{S}}{\rho}\right)^{water}_{material}$ varies less than one percent throughout a photon radiation field for a given photon beam energy. Therefore, a simple material dependent post-processing multiplicative factor can be used to convert $D_{material}$ to D_{water} for a given photon beam energy.

VMC++ converts $D_{material}$ to D_{water} during execution of the transport on a track-by-track basis, not as a post-processing step. As the electron is transported through the geometry, Bragg-Gray cavity theory is utilized to compute D_{water} by multiplying the energy deposited by the ratio of stopping powers. In VMC++ all energy-dependent quantities are evaluated at a randomly distributed energy between the initial and final energies of the step. In the case of computing D_{water}, therefore, the energy deposition by an electron track in a voxel is multiplied by the ratio of stopping powers evaluated at a

random energy between that of the electron entering the voxel and that of the electron exiting the voxel.

To compare the D_{water} computation methods, the same patient plan described in section 5.4 was used. VMC++ was run to produce a dose distribution in terms of D_{water}. VMC++ was also run to produce a dose distribution in terms of $D_{material}$. The resulting $D_{material}$ was converted to D_{water} using the in-house algorithm. The two distributions were then compared using gamma analysis for voxels with dose greater than fifty percent of the maximum dose. Also, the distributions were compared using the Kawrakow-Fippel method. Finally, respective DVHs were produced.

Results of the gamma analysis are seen in Table 6. Whereas 99.8% of the voxels passed the 2% / 2 mm criteria, 99.5% passed the 1% / 1mm criteria. It should be noted that these results lie within the bounds of statistical uncertainty calculated in section 6.1. MATLAB was used to fit the data for the Kawrakow-Fippel method results. The fit may be seen in Figure 28. According to the fit, a fraction $\alpha_1 = 1.0$ of the voxels have a systematic difference of $\delta_1 = 0.17$ standard deviations. The value for α_2 corresponded to zero. The VMC++ internal D_{water} calculation seems to slightly underpredict the VCU in-house conversion. However, since the standard deviation in this case refers to a combined nominal statistical uncertainty of 0.7% of the two calculations, the difference corresponds to a 0.1% difference of the maximum dose. This difference is insignificant compared to the systematic error introduced by the in-house algorithm, which approximates that $(\frac{\overline{S}}{\rho})_{material}^{water}$ does not vary throughout a photon radiation field, when in fact it does by up to one percent. The comparison DVH, seen in Figure 29, shows that the differences between the two methods are visually indiscernible. The D_{90} of the CTVs agrees within 0.1 percent.

8 Conclusion & Future Work

VMC++ was integrated into a commercial TPS at VCU and benchmarked against DOSXYZnrc. Output factor normalization was performed, yielding agreement within 1% of experiment for all field sizes except 1x1 cm^2. Geometric validation was carried out on an irregular water phantom, and showed that VMC++ properly aligned the beam on phantom geometry. VMC++ dose calculations were compared to DOSXYZnrc dose calculations for a simple water phantom and a complex nine-field IMRT H/N patient case. The VMC++ dose distribution differed insignificantly with that of DOSXYZnrc for the water phantom case. For the patient case, 5.28% of the VMC++ voxels deviated from the DOSXYZnrc voxels by 4.06 standard deviations. Considering the combined statistical uncertainty of 0.7%, this resulted in a 2.8% difference of maximum dose. This result warrants further investigation (see following paragraph) since the difference between the two codes exceeds the proposed criteria of ICRU report 42[36] of 2% of maximum dose. When the gamma index was plotted in the patient geometry, it was seen that the voxels which failed the gamma analysis were located mostly in air cavities within the body. The internal dose-to-water conversion of VMC++ was investigated by comparing it to a VCU in-house conversion. The two resulting dose distributions strongly agreed, with 99.5% of the voxels passing the gamma analysis within 1% / 1mm criteria, and the resulting differences in DVHs were visually indiscernible.

Future investigation should be performed concerning the discrepancy between the dose distributions calculated by DOSXYZnrc and VMC++ for the H/N patient case. A proposed flow chart for this investigation is seen in Figure 30. Results of this thesis show that DOSXYZnrc and VMC++ agree within statistical uncertainty for the simple case of a 10x10 cm^2 field orthogonally incident on a water phantom. The next step would be to test if the presence of materials other than water is causing disagreement between the two codes. This test ('test A') would be accomplished by having a 10x10 cm^2 field orthogonally incident on a heterogeneous phantom. If test A were to fail, the next step ('test B') would be to change the CT ramp from the 56-material ramp to the simplistic CT

ramp that was predefined in VMC++. If test B were to fail, it could be concluded that there is a problem with implementation of VMC++, since previous studies have shown that DOSXYZnrc and VMC++ agree in a test case similar to test B.[21] If test B were to show agreement, it could be concluded that there is a problem in the implementation of the 56-material CT ramp. If test A were to show agreement, the next step ('test C') would be to test if the presence of field sizes besides 10x10 cm^2 is causing the discrepancy. This would be accomplished by simulating one beam from the H/N patient plan at 0^o on a water phantom, except with the IMRT turned off. If test C were to fail, it could be concluded that there is a problem in the implementation of field sizes. If test C were to show agreement, the next step ('test D') would be to rerun test C with IMRT turned on. If test D were to fail, it could be concluded that there is a problem in the implementation of IMRT, perhaps how VMC++ is reading in the weights of the incident particles. If test D were to show agreement, the next step ('test E') would be to test if the presence of various incident beam angles besides 0^o is causing the discrepancy. Test E could be performed by simulating the entire H/N patient plan on a water phantom. If test E were to fail, the next step ('test F') would be to perform a beam-by-beam analysis of all the beams in the H/N patient plan. If test F were to result in significantly different levels of agreement among all the beams, the next step would be to determine what differentiates one beam from the next. If test E were to show agreement, the next step ('test G') would be to test if the presence of the patient's outer contour is causing the discrepancy by replacing the water phantom with the patient geometry composed entirely of water. If test G were to fail, it could be concluded that the patient's outer contour is causing the discrepancy. If test G were to show agreement, the next step ('test H') would be to test if the patient materials and internal geometry are causing the discrepancy. Test H would be accomplished by calculating the dose distributions for the H/N patient plan on the H/N patient geometry and materials. If test H were to fail, the next step ('test I') would be to test a prostate plan on patient geometry to observe if the discrepancy is specific to the H/N patient plan. If test I were to fail, it could be concluded that the combination of patient materials with inner geometry is causing the discrepancy. If test I

were to show agreement, it could be concluded that the discrepancy is H/N patient geometry specific. Finally, if test H were to show agreement, the discrepancy would be resolved. To expedite the testing process, the tests should not be performed in order, but instead each subsequent test should be halfway between the tests with known results.

Figures

Table 1: Summary of tests for geometric validation.

Test	Reasoning
Unique gantry angle	Validate that VMC++ is positioning the gantry angle correctly.
Unique collimator angle	Validate that VMC++ is positioning the collimator angle correctly.
Unique couch angle	Validate that VMC++ is positioning the couch angle correctly.
Unique gantry, collimator, and couch angles	Validate that VMC++ is positioning all angle correctly without introducing error due to angle coupling.
Change the SSD	Validate that VMC++ correctly places the phase space of incident particles at the proper location.

Table 2: Definitions of the gamma function.[32]

Symbol	Equation	Description
$D_e(\vec{r}_e)$	N/A	Evaluated dose D_e at position $\vec{r}_e$
$D_r(\vec{r}_r)$	N/A	Reference dose D_r at position $\vec{r}_r$
ΔD	N/A	Dose difference criterion
Δd	N/A	Distance-to-agreement criterion
$r(\vec{r}_e,\vec{r}_r)$	$r(\vec{r}_e,\vec{r}_r)=\lvert\vec{r}_e-\vec{r}\rvert$	Spatial distance between evaluated and reference dose points
$\delta(\vec{r}_e,\vec{r}_r)$	$\delta(\vec{r}_e,\vec{r}_r)=D_e(\vec{r}_e)-D_r(\vec{r}_r)$	Difference between evaluated dose $D_e(\vec{r}_e)$ at position $\vec{r}_e$ and reference dose $D_r(\vec{r}_r)$ at position $\vec{r}_r$
$\Gamma(\vec{r}_e,\vec{r}_r)$	$\Gamma(\vec{r}_e,\vec{r}_r)=\sqrt{\frac{r^2(\vec{r}_e,\vec{r}_r)}{\Delta d^2}+\frac{\delta^2(\vec{r}_e,\vec{r}_r)}{\Delta D^2}}$	Generalized Γ function, computed for all evaluated positions $\vec{r}_e$ and reference positions $\vec{r}_r$
$\gamma(\vec{r}_r)$	$\gamma(\vec{r}_r)=\min\{\Gamma(\vec{r}_e,\vec{r}_r)\}\forall\{\vec{r}_e\}$	γ function, the minimum generalized Γ function in the set of evaluated points

Table 3: Gamma analysis results for comparison between two VMC++ dose distributions for a 6MV 10x10cm^2 beam incident on a water phantom. Only voxels with dose greater than fifty percent of maximum dose were included in the analysis.

Dose difference/DTA Criteria	Percent passed	Mean gamma value
2% / 2mm	99.992	0.135
1% / 1mm	99.296	0.264

Table 4: Gamma analysis results for comparison between DOSXYZnrc and VMC++ dose distributions for a 6MV 10x10cm^2 beam incident on a water phantom. Only voxels with dose greater than fifty percent of maximum dose were included in the analysis.

Dose difference/DTA Criteria	Percent passed	Mean gamma value
2% / 2mm	100	0.144
1% / 1mm	98.632	0.280

Table 5: Gamma analysis results for comparison between DOSXYZnrc and VMC++ dose distributions for a nine-beam IMRT plan on a head-and-neck (H/N) patient. Only voxels with dose greater than fifty percent of maximum dose were included in the analysis.

Dose difference/DTA Criteria	Percent passed	Mean gamma value
3% / 3mm	99.984	0.198
2% / 2mm	99.074	0.262
1% / 1mm	86.350	0.523

Table 6: Gamma analysis results for comparison between VMC++ $D_{material}$ plus an in-house conversion to D_{water} and VMC++ D_{water} for a nine-beam IMRT plan on a H/N patient. Only voxels with dose greater than fifty percent of maximum dose were included in the analysis.

Dose difference/DTA Criteria	Percent passed	Mean gamma value
2% / 2mm	99.788	0.155

1% / 1mm	99.477	0.200

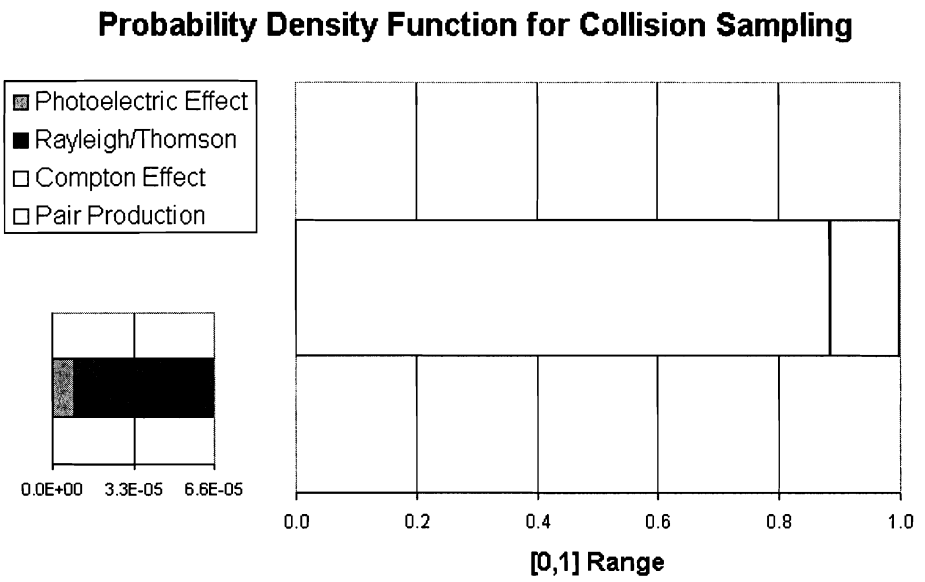

Figure 1: Probability density function for collision sampling. In this case, a 6 MeV photon is traveling through a water medium. The computer will randomly sample a number on the range [0,1]. The most probable interaction is a Compton effect (yellow band), which will be sampled roughly 88% of the time. Pair production (light blue band) will be sampled about 12% of the time. The less probably cross-sections are seen in the inset located at the lower left side of the figure. Even though the cross-sections for photoelectric effect (blue-purple band) and Rayleigh/Thomson scattering (maroon band) are on the order of 10^{-5}, these events will be selected when running Monte Carlo for millions of particle histories.

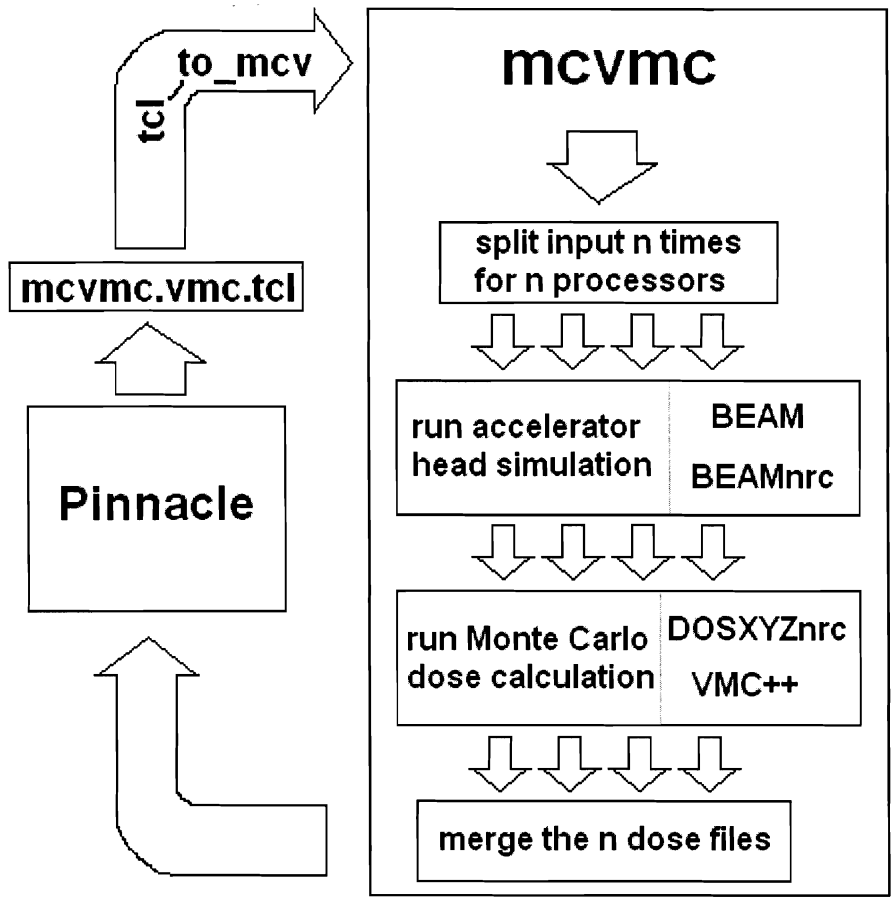

Figure 2: Basic flow diagram for the mcvmc process. The user selects the mcvmc.vmc.tcl script to run mcvmc. The user-specified flags and values are passed to mcvmc by tcl_to_mcv. The input is split n times to be divided among processors. The user-specified accelerator head simulation is carried out followed by the user-specified Monte Carlo dose calculation. After the dose files are merged together, the dose is displayed in Pinnacle. It should be noted that this is a simplified flow diagram, which does not include other available options such as running the particles through the dynamic multileaf collimator or using an electronic portal imaging device after transport through the phantom geometry.

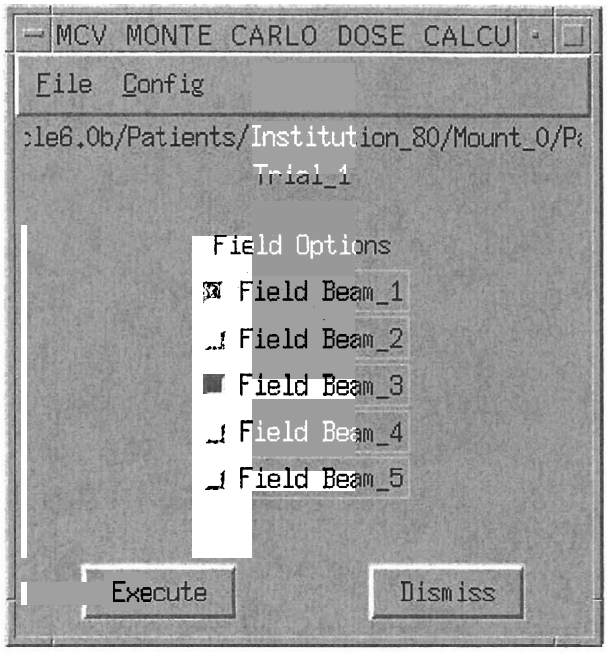

Figure 3: GUI resulting from mcvmc.vmc.tcl for a five beam treatment plan (beams named Beam_1, Beam_2, ...; trial named Trial_1). In this case, particle transport and dose deposition will be performed for Beam_1 and Beam_3. To set other options, the different Config menus must be accessed.

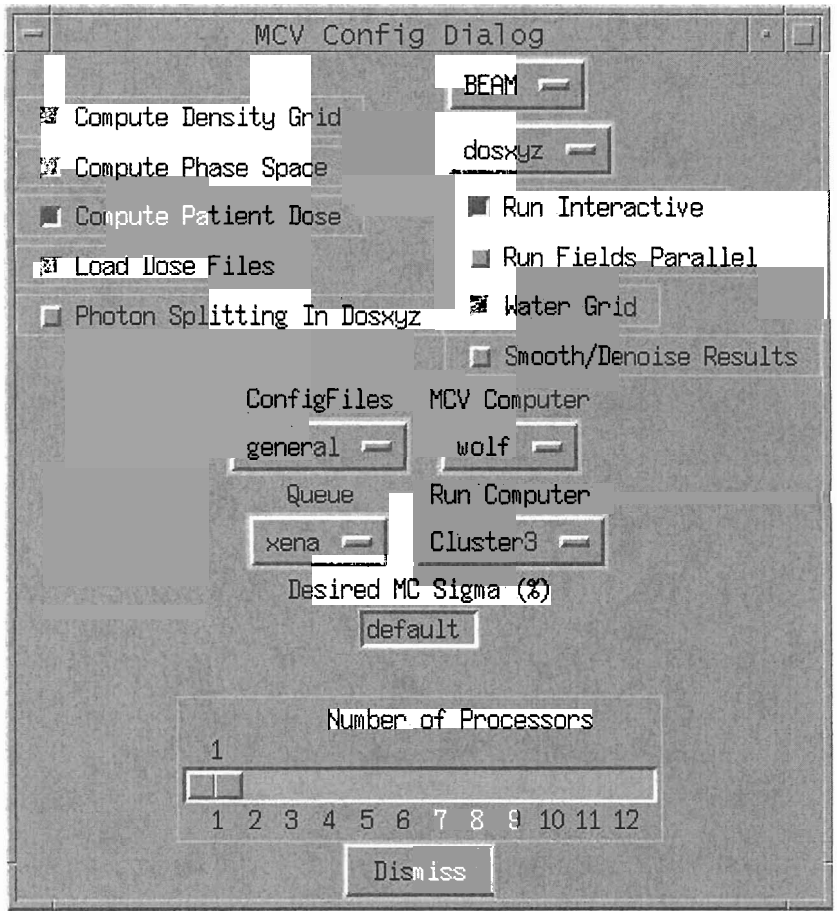

Figure 4: The general config menu created by mcvmc.vmc.tcl. The user can choose the desired Monte Carlo code to use (dosxyz, dosxyznrc, mcvrtp, vmc++) by clicking on the second tab at the top right.

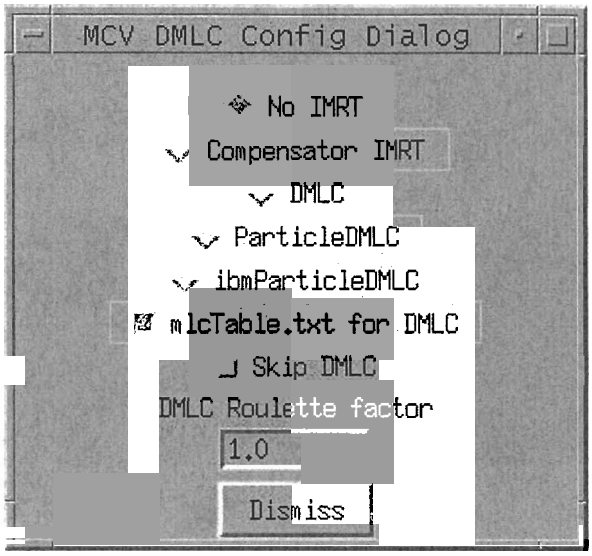

Figure 5: The DMLC config menu created by mcvmc.vmc.tcl. The user can specify whether or not IMRT is being used, and which code to run the particles through the multi-leaf collimator.

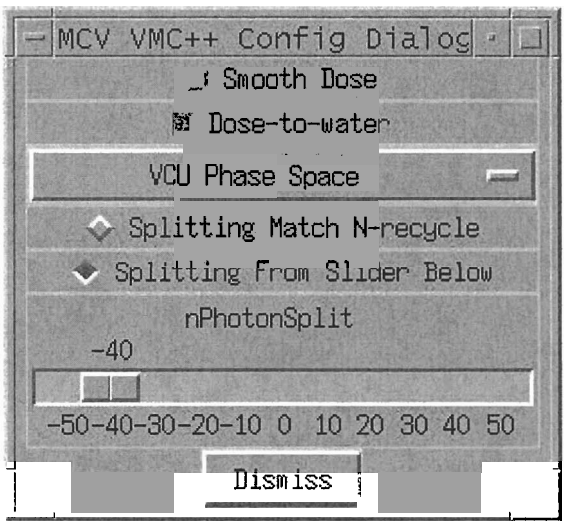

Figure 6: The VMC++ config menu created by mcvmc.vmc.tcl. Dose-to-water or dose-to-material may be chosen. The user can also choose what type of simulation (source model or phase space) to use, as well as if they wish to match photon splitting from DOSXYZnrc or set it independently.

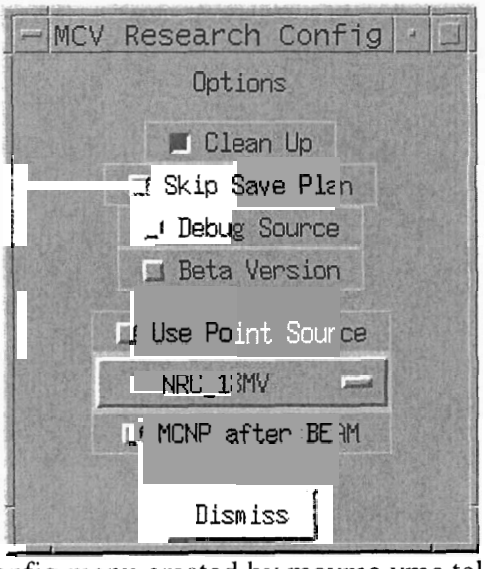

Figure 7: The research config menu created by mcvmc.vmc.tcl. The user may choose to use new particles for transport by checking the 'Clean Up' option. 'Beta Version' may be checked to test new code safely.

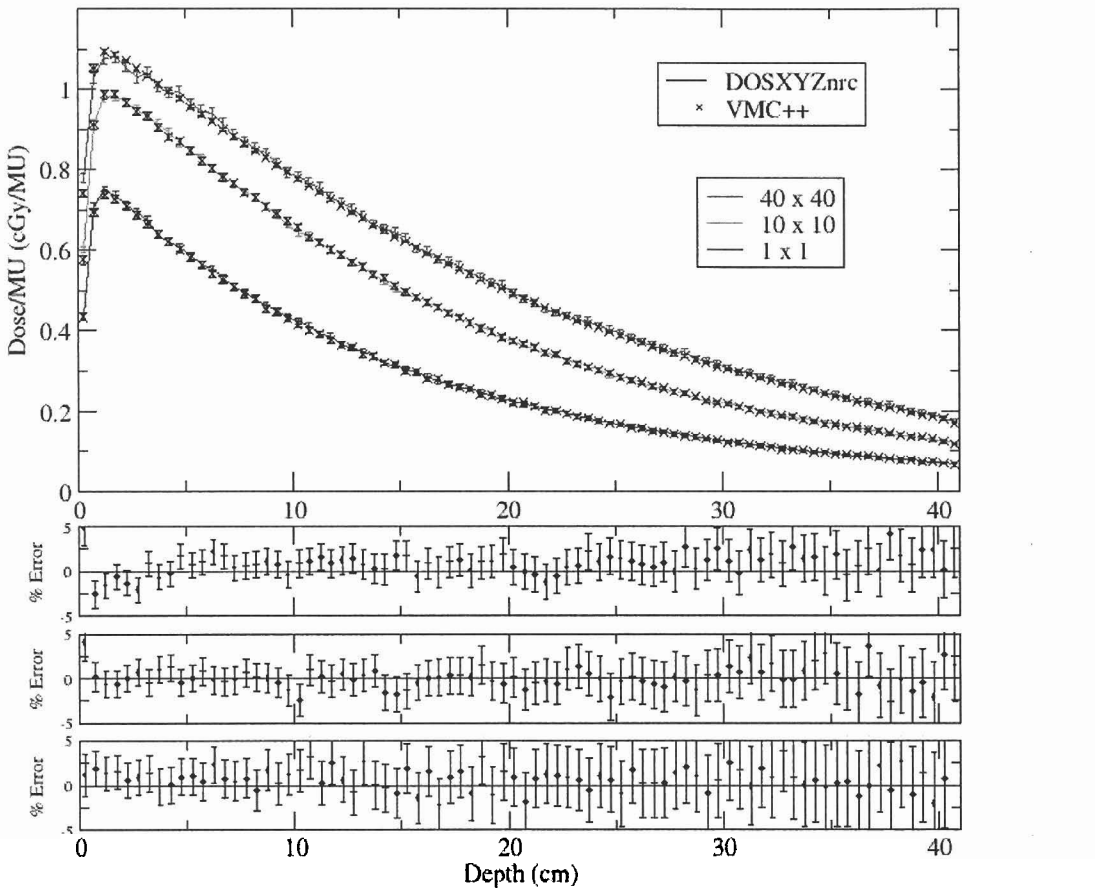

Figure 8: Comparison of DOSXYZnrc and VMC++ depth doses (cGy/MU) for 40x40 cm^2, 10x10 cm^2, and 1x1 cm^2 fields at energy 6MV. The lower plots are the respective percent error plots, calculated using the formula: $\%Error = (D_{DOSXYZnrc} - D_{VMC++}) / D_{DOSXYZnrc} * 100\%$, since DOSXYZnrc was considered the standard with which to compare VMC++. Statistical uncertainty was propagated by summing by quadrature. The error bars in all plots correspond to two times the standard deviation.

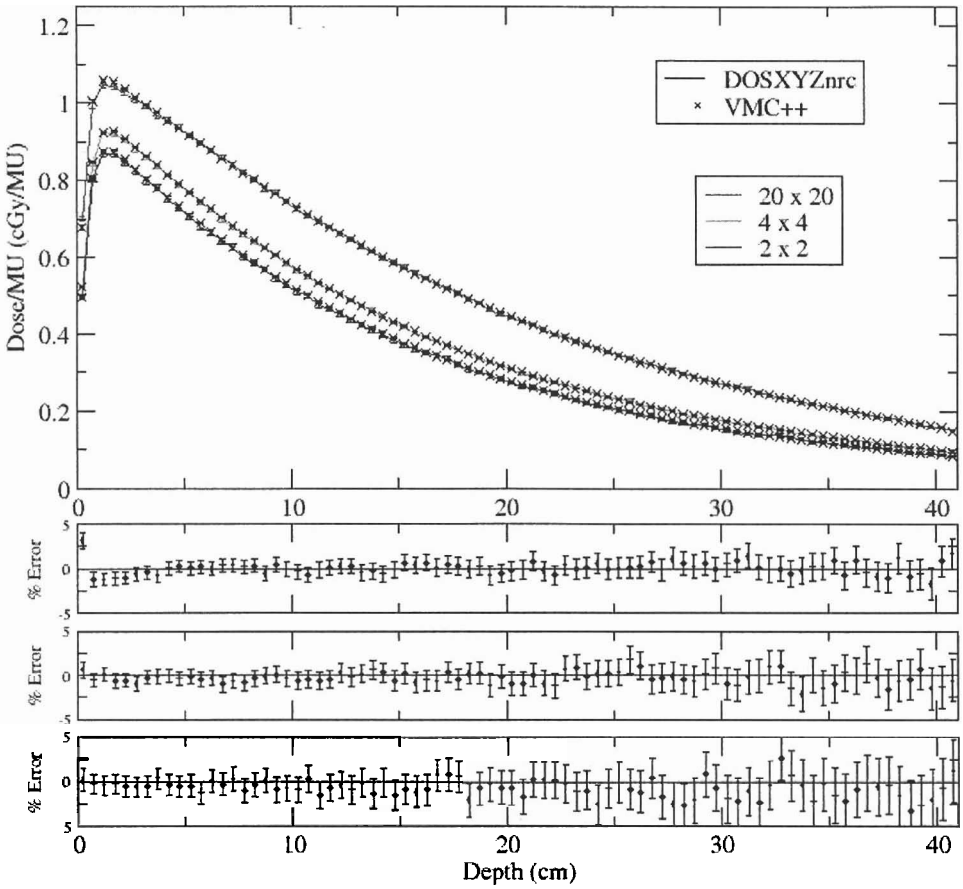

Figure 9: Comparison of DOSXYZnrc and VMC++ depth doses (cGy/MU) for 20x20 cm^2, 4x4 cm^2, and 2x2 cm^2 fields at energy 6MV. The lower plots are the respective percent error plots, calculated using the formula: $\%Error = (D_{DOSXYZnrc} - D_{VMC++}) / D_{DOSXYZnrc} * 100\%$. Statistical uncertainty was propagated by summing by quadrature. The error bars in all plots correspond to two times the standard deviation.

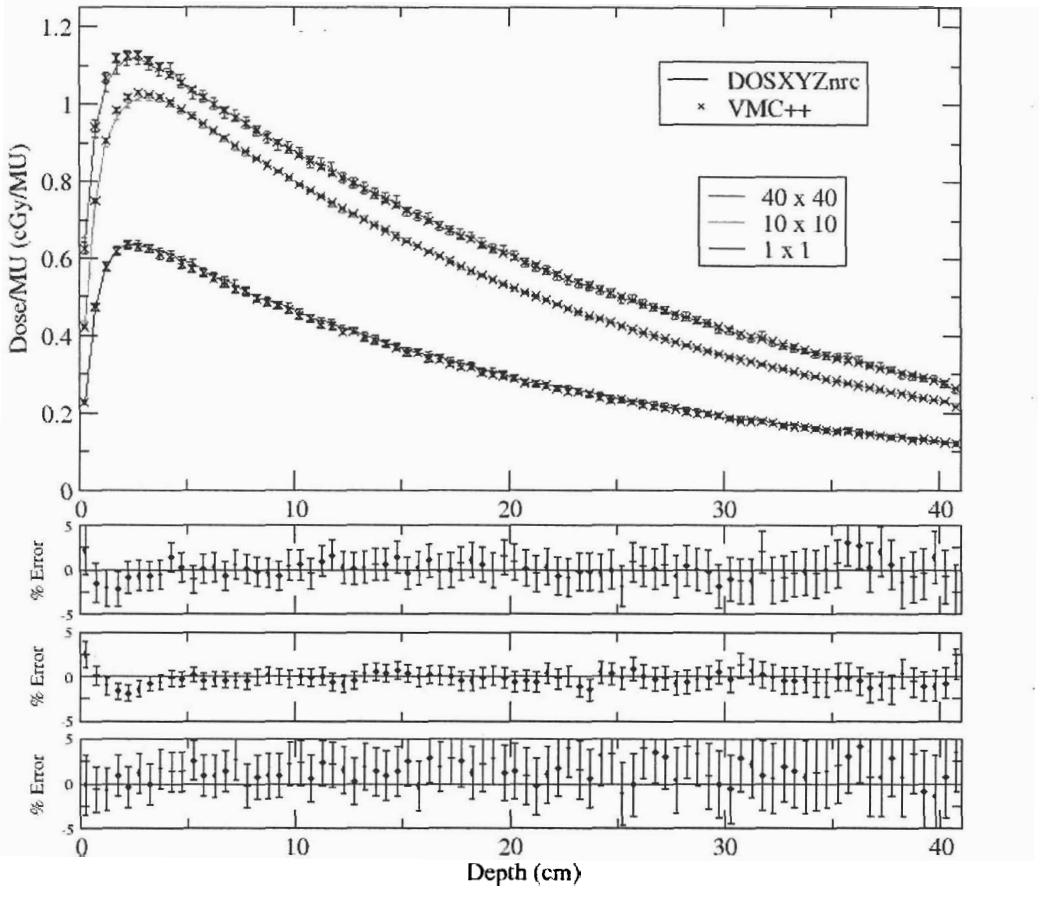

Figure 10: Comparison of DOSXYZnrc and VMC++ depth doses (cGy/MU) for 40x40 cm^2, 10x10 cm^2, and 1x1 cm^2 fields at energy 18MV. The lower plots are the respective percent error plots, calculated using the formula: $\%Error = (D_{DOSXYZnrc} - D_{VMC++}) / D_{DOSXYZnrc} * 100\%$. Statistical uncertainty was propagated by summing by quadrature. The error bars in all plots correspond to two times the standard deviation.

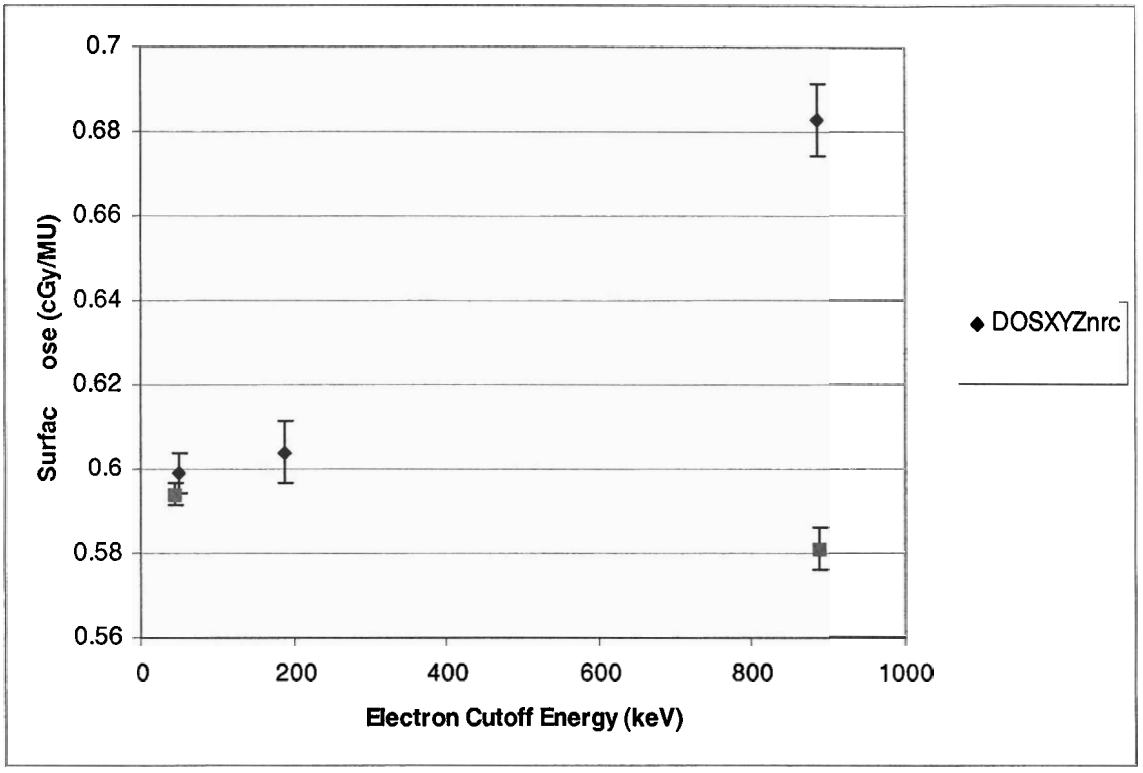

Figure 11: Comparison of DOSXYZnrc and VMC++ surface doses (cGy/MU) for 10x10 cm^2 field at energy 6MV. The surface doses calculated by the two codes converge as the electron cutoff energy decreases. The error bars correspond to two standard deviations.

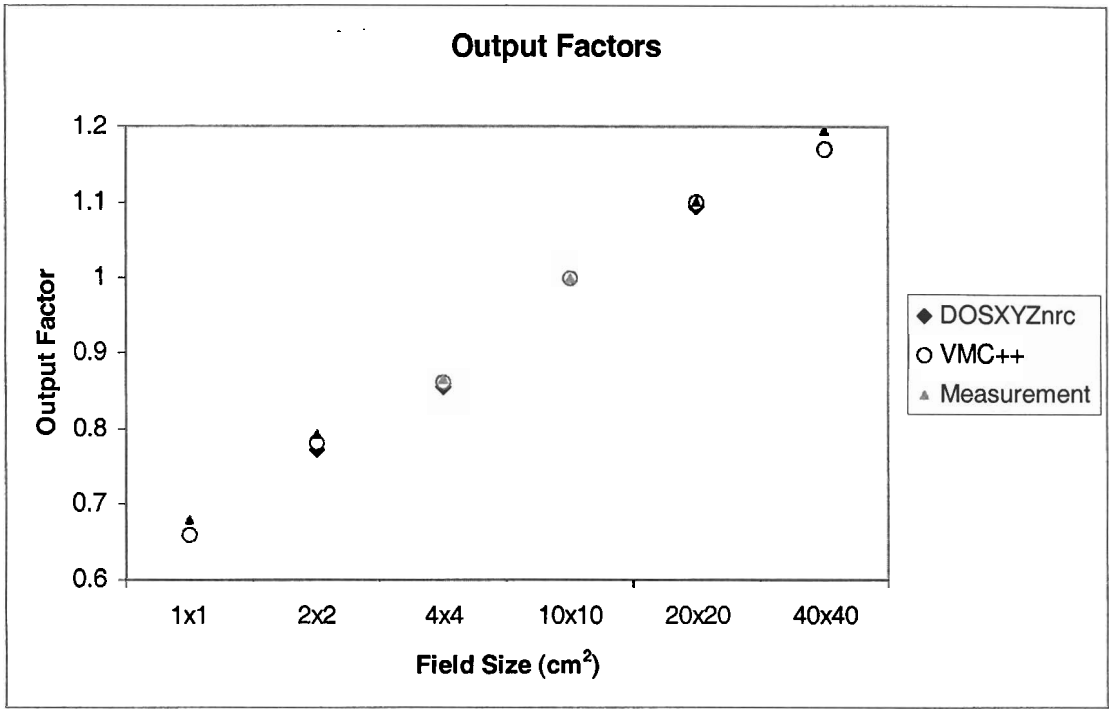

Figure 12: Output factor comparison between DOSXYZnrc, VMC++, and measurement for various field sizes and energy 6MV. The difference in the 1x1 cm^2 field may be attributed to the uncertainty involved in the positioning of the dosimeter in experiment.

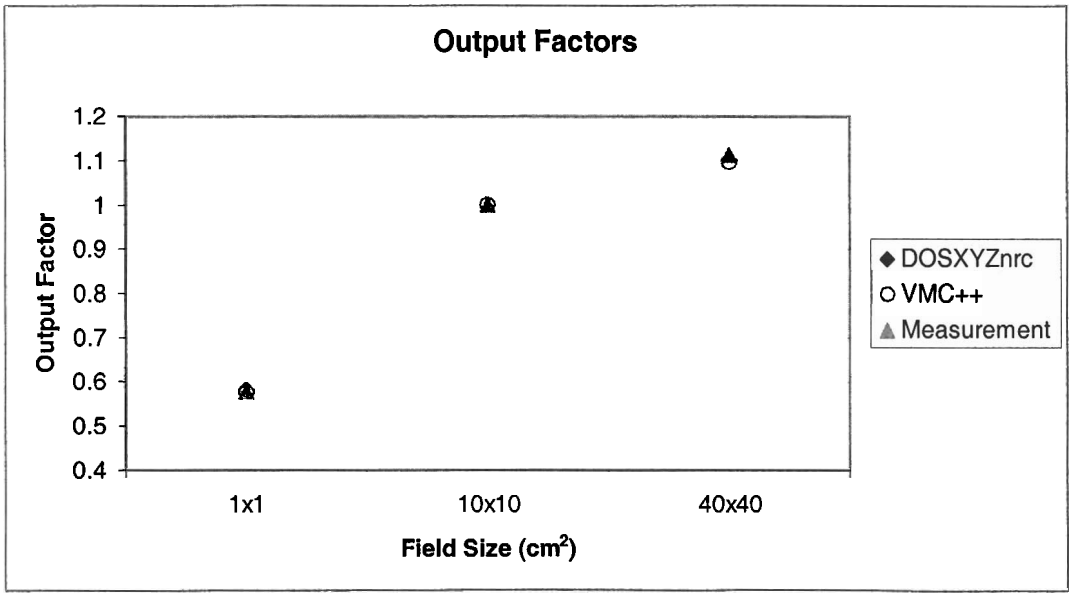

Figure 13: Output factor comparison between DOSXYZnrc, VMC++, and measurement for various field sizes and energy 18MV.

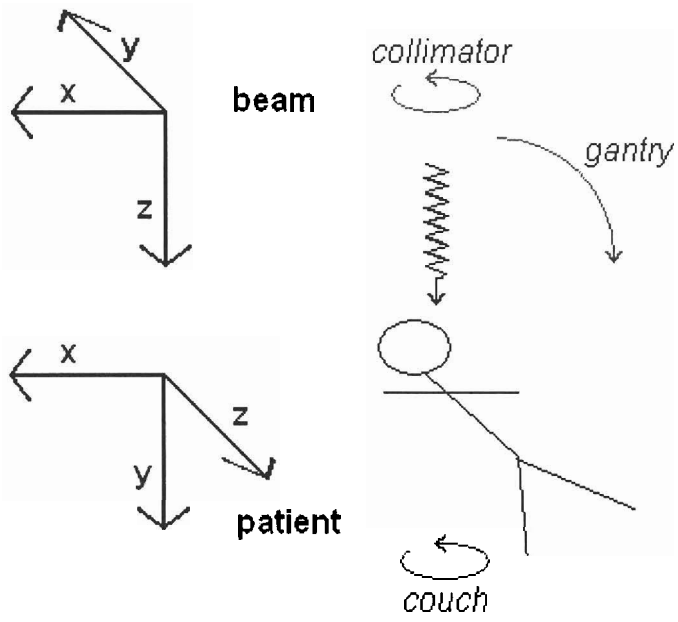

Figure 14: Beam and patient coordinate systems. In order to transform from beam coordinates to patient coordinates, each particle's position and direction must be multiplied by a rotation matrix to properly account for gantry, collimator, and couch rotations.

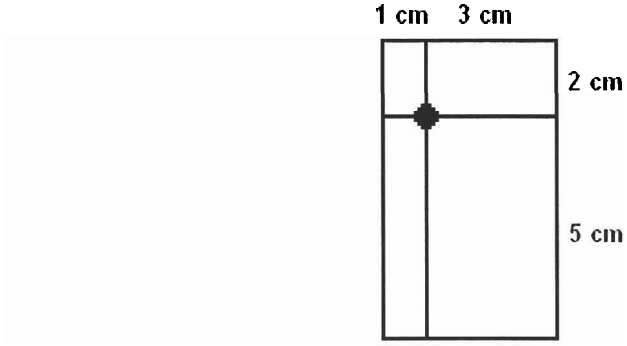

Figure 15: Irregular field shape used for phantom geometric validation. The black dot shows the beam axis.

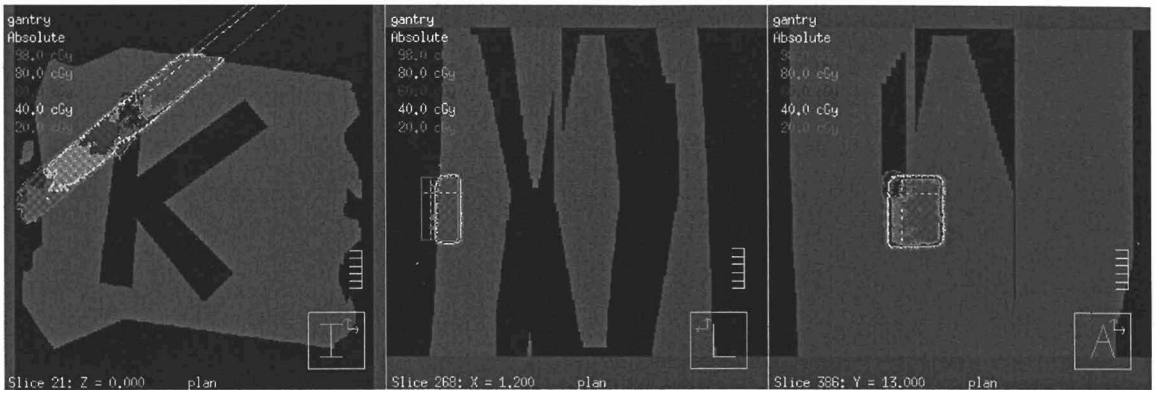

Figure 16: Transverse, sagittal, and coronal view of dose distribution computed by VMC++ for a gantry angle of 53°. Note how the distribution lines up with the projected aperture. Also, note the dose fall-off when the beam encounters air cavities within the phantom.

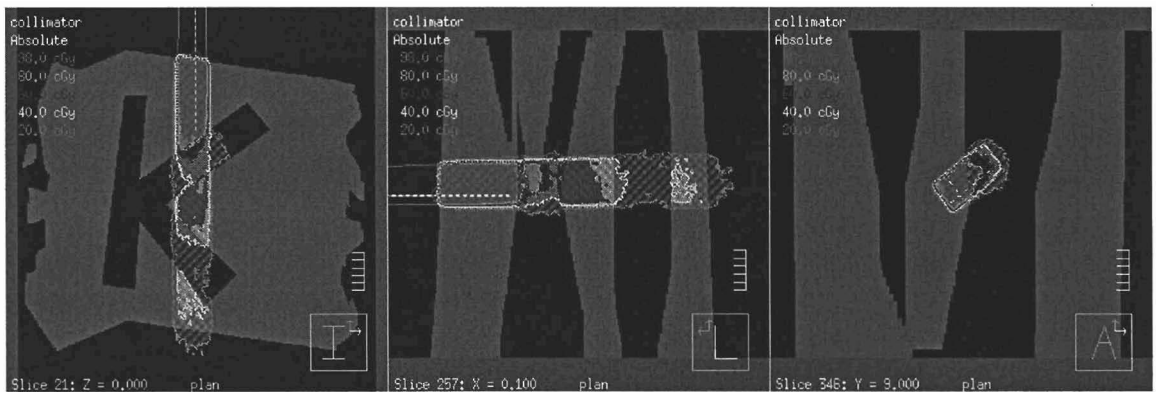

Figure 17: Transverse, sagittal, and coronal view of dose distribution computed by VMC++ for a collimator angle of 128°. Note how the distribution lines up with the projected aperture. Also, note the dose fall-off when the beam encounters air cavities within the phantom.

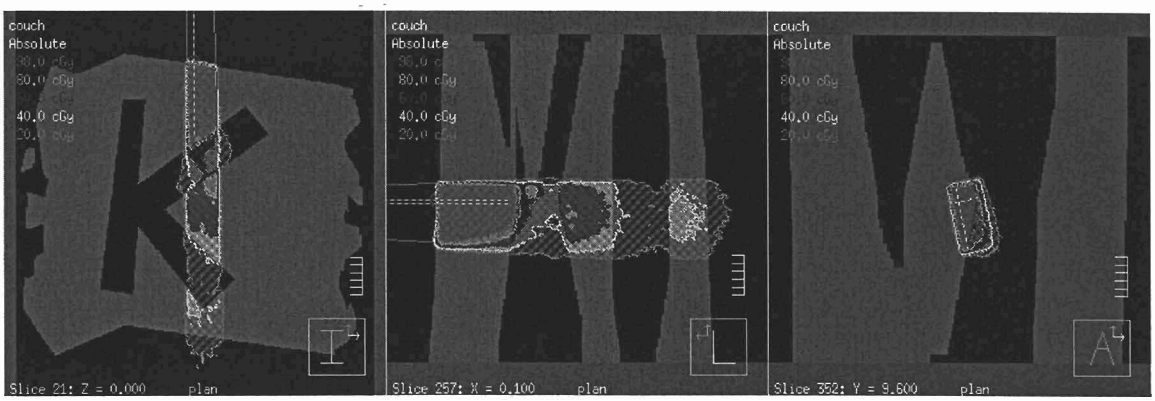

Figure 18: Transverse, sagittal, and coronal view of dose distribution computed by VMC++ for a couch angle of 12°. Note how the distribution lines up with the projected aperture. Also, note the dose fall-off when the beam encounters air cavities within the phantom.

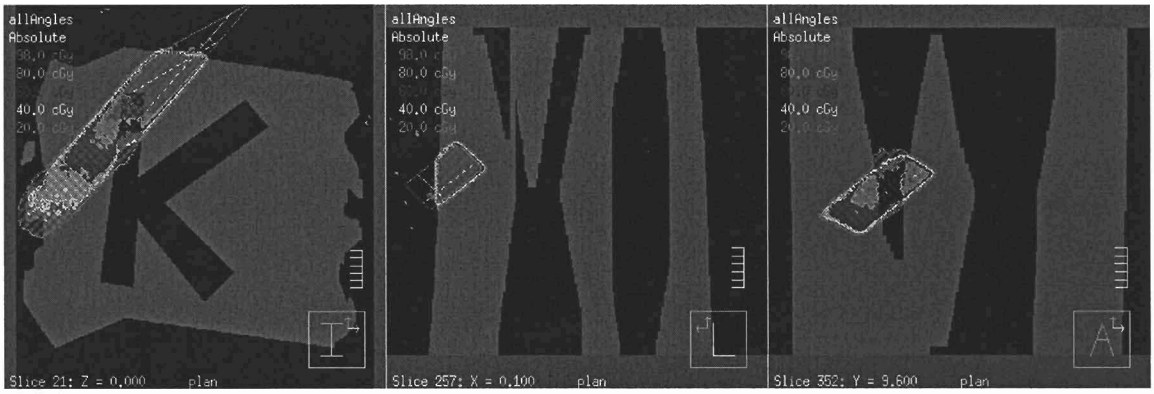

Figure 19: Transverse, sagittal, and coronal view of dose distribution computed by VMC++ for a gantry angle of 53°, collimator angle of 128°, and couch angle of 12°. Note how the distribution lines up with the projected aperture. Also, note the dose fall-off when the beam encounters air cavities within the phantom.

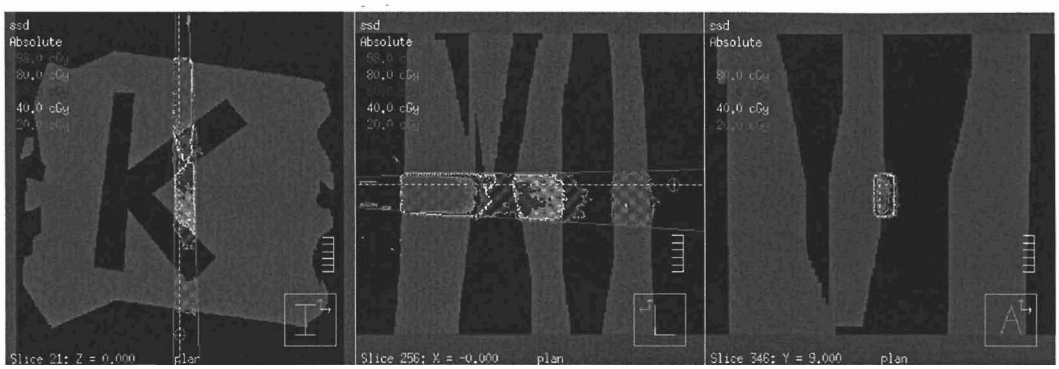

Figure 20: Transverse, sagittal, and coronal view of dose distribution computed by VMC++ for an SSD of 60cm. Note how the distribution lines up with the projected aperture. Also, note the dose fall-off when the beam encounters air cavities within the phantom.

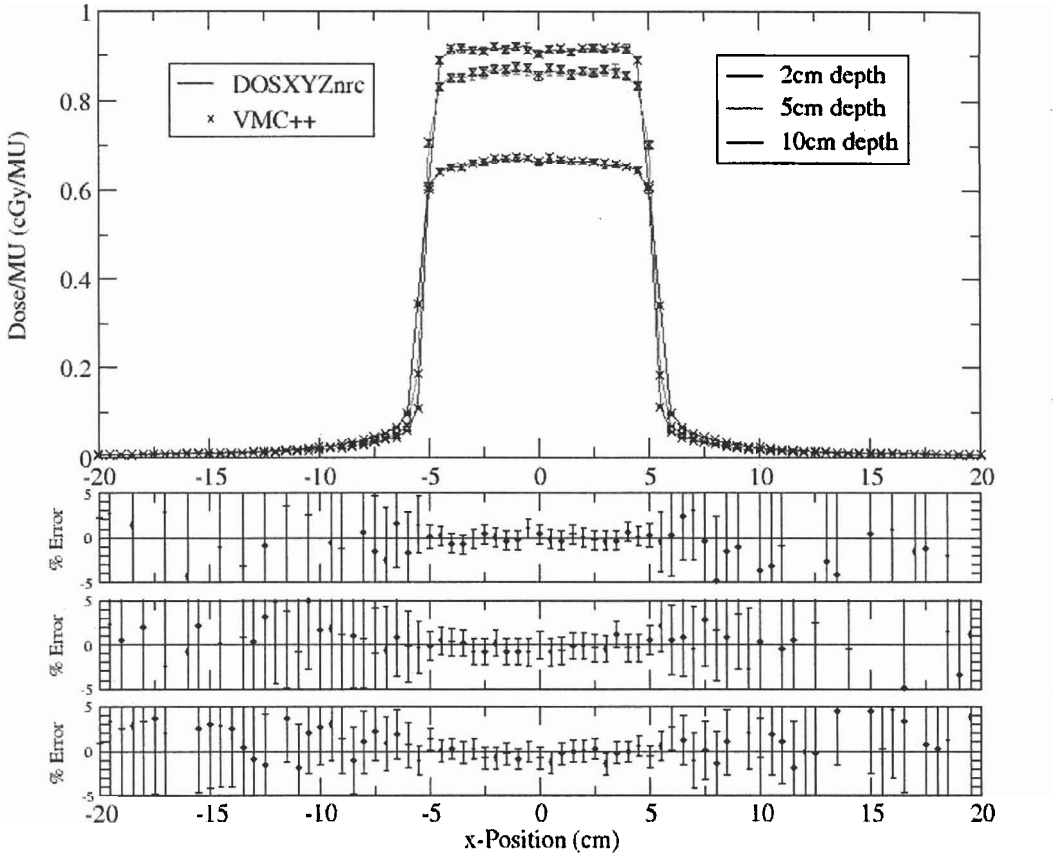

Figure 21: Comparison of DOSXYZnrc and VMC++ lateral profiles in the x-direction of the patient coordinate system for a 10x10 cm^2 field at energy 6MV. The lower plots are the respective percent error plots, calculated using the formula: $\%Error = (D_{DOSXYZnrc} - D_{VMC++}) / D_{DOSXYZnrc} * 100\%$. Statistical uncertainty was propagated by summing by quadrature. The error bars in all plots correspond to two times the standard deviation.

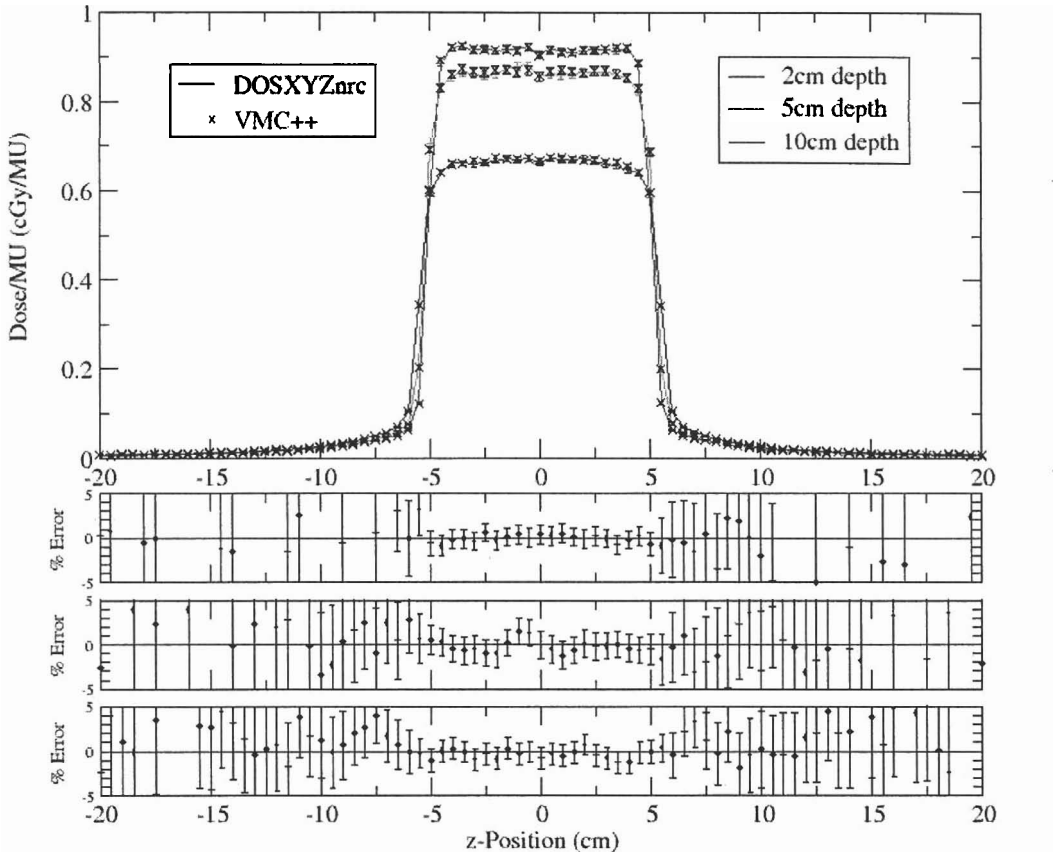

Figure 22: Comparison of DOSXYZnrc and VMC++ lateral profiles in the z-direction of the patient coordinate system for a 10x10 cm^2 field at energy 6MV. The lower plots are the respective percent error plots, calculated using the formula: $\%Error = (D_{DOSXYZnrc} - D_{VMC++}) / D_{DOSXYZnrc} * 100\%$. Statistical uncertainty was propagated by summing by quadrature. The error bars in all plots correspond to two times the standard deviation.

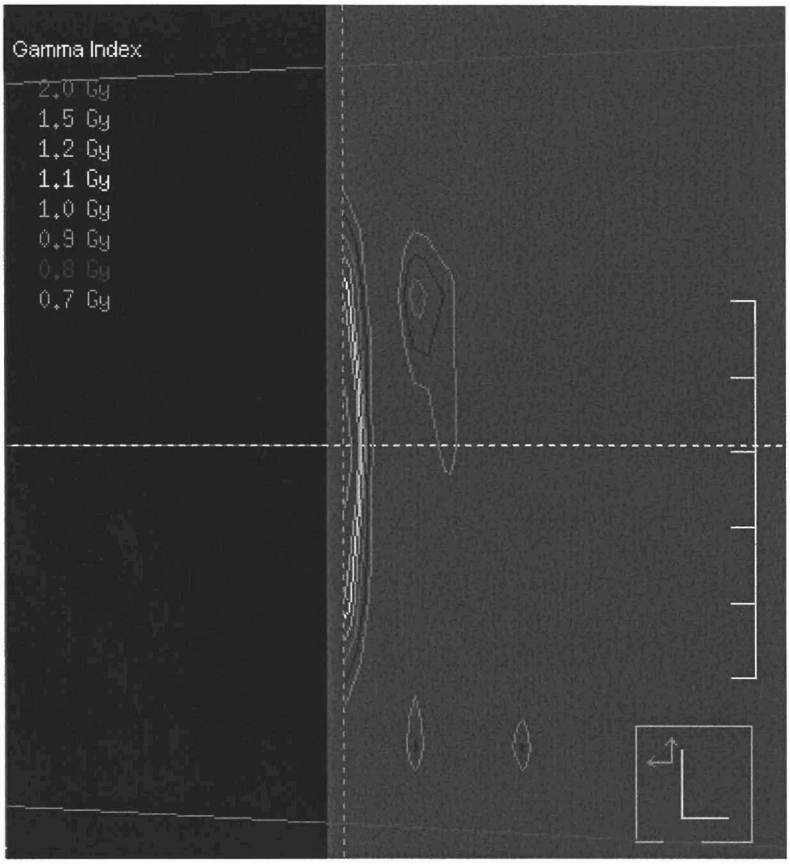

Figure 23: Gamma index results displayed on top of the water phantom. The beam axis is shown by the dashed red line. Gamma index criteria were 1% / 1mm in this case. The only area where the gamma index was greater than one was on the front face of the phantom. Note that the misalignment between the phantom surface and the dose grid (dashed green line) is a visual quirk of Pinnacle: mathematically, the two line up exactly.

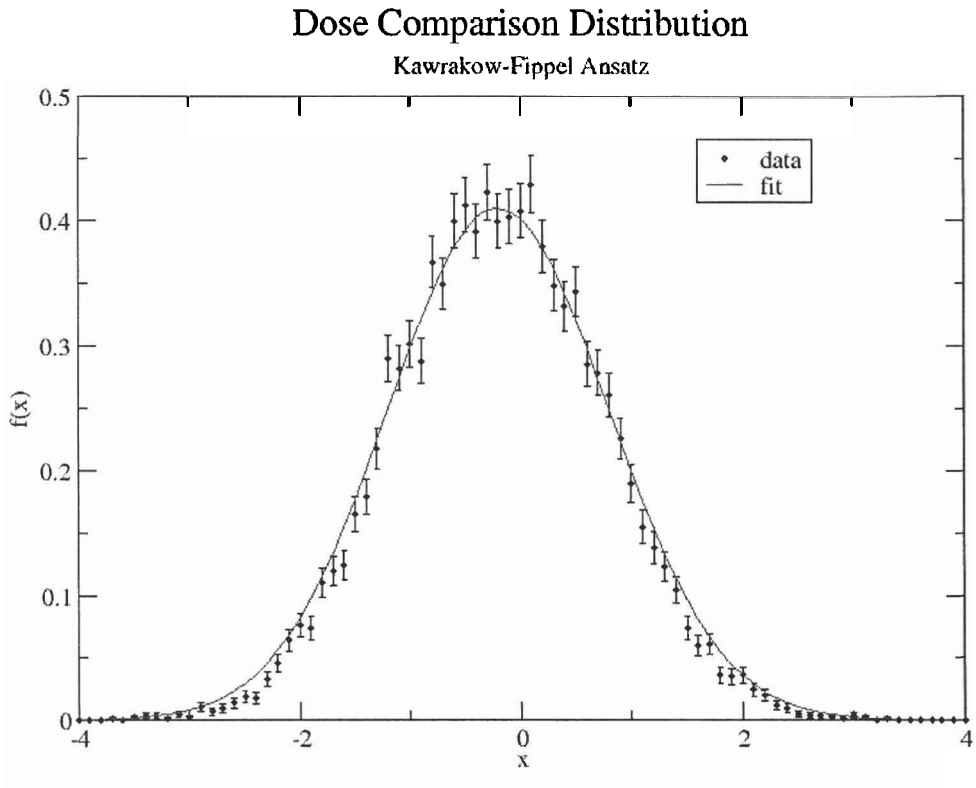

Figure 24: Fit of the dose distribution comparison. The fit determined that 60% of the voxels had no systematic deviation between the two codes, while the remaining 40% of the voxels had a difference of 0.27 standard deviations. In this case the standard deviation is equal to the combined nominal statistical uncertainty, 0.7%. Therefore, the difference between the two distributions (0.2%) is insignificant for the differing 40% of the voxels. The reason the fit seems slightly too broad is due to the lack of accounting for cross-correlation when calculating the combined statistical uncertainty in each voxel.

Figure 25: Gamma index displayed on top of a H/N patient for which a nine-beam IMRT plan was used. Criteria were 2% / 2mm in this case. The only areas where the gamma index was greater than one were air cavities in the patient.

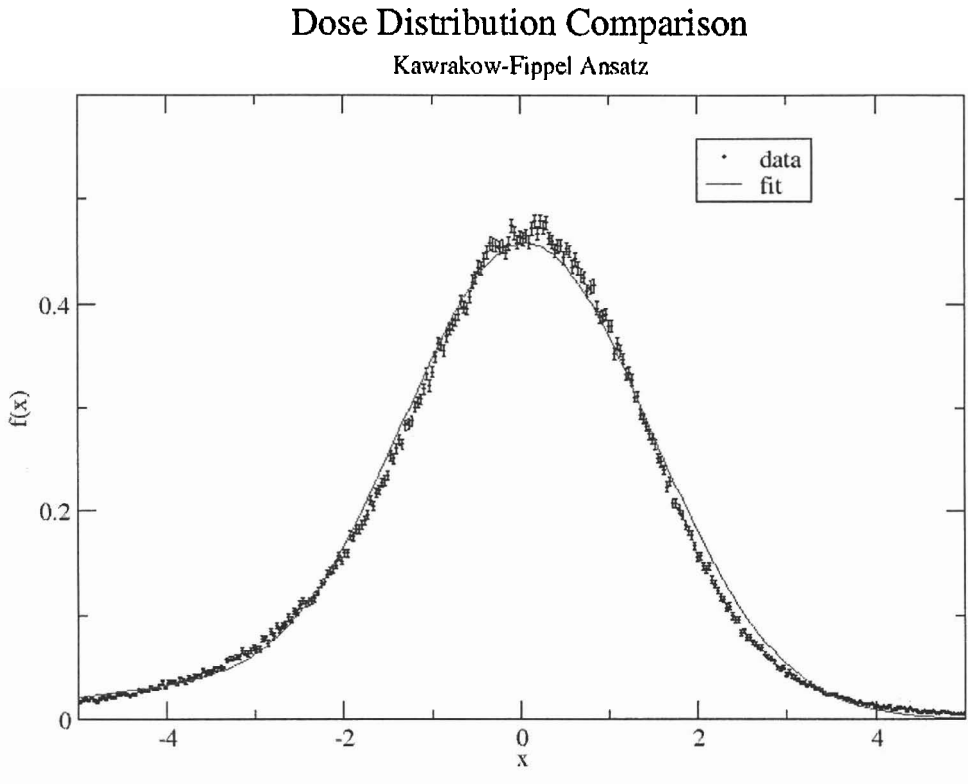

Figure 26: Fit of the dose distribution comparison. The fit determined that approximately 95% of the voxels had no standard deviation between the two codes, while the remaining 5% of the voxels had a difference of 4.06 standard deviations. In this case the standard deviation is equal to the combined nominal statistical uncertainty, 0.7%. Therefore, the difference between the two distributions is 2.8% for the differing 5% of the voxels. The reason the fit seems slightly too broad is due to the lack of accounting for cross-correlation when calculating the combined statistical uncertainty in each voxel.

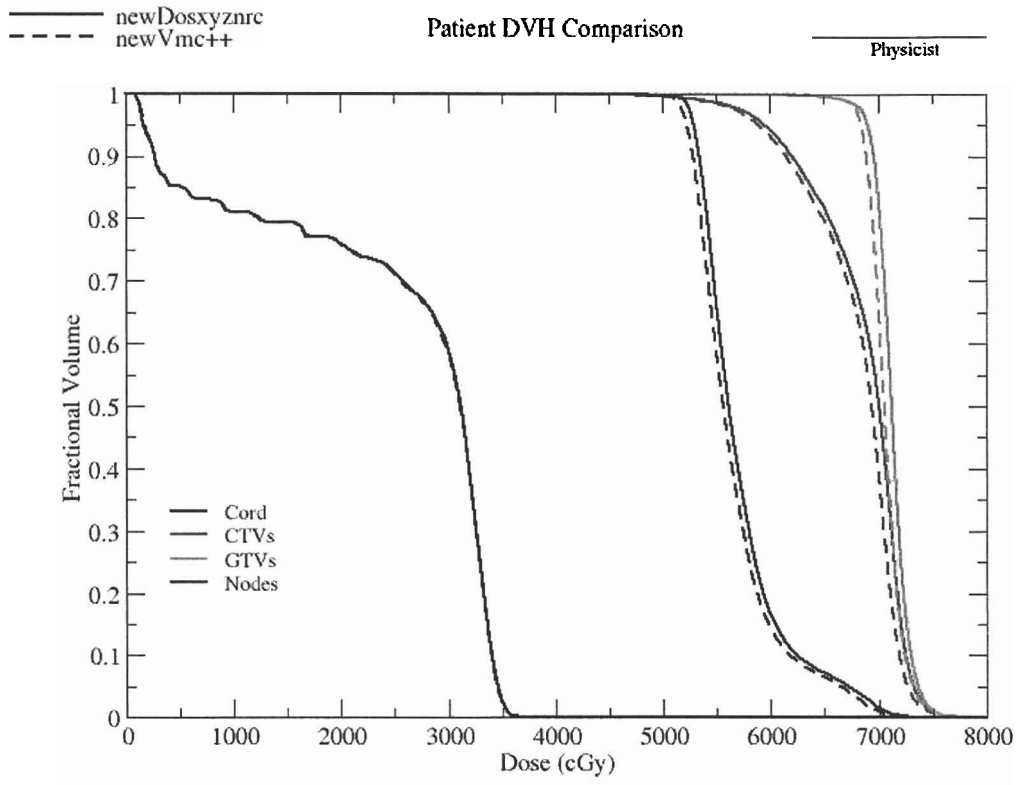

Figure 27: DVH comparing DOSXYZnrc dose calculation (solid lines) with VMC++ (dashed lines) for the spinal cord (black), the nodes (blue), clinical target volumes (CTVs) (red), and gross tumor volumes (GTVs) (green). VMC++ noticeably underpredicts DOSXYZnrc for the GTVs, CTVs, and nodes.

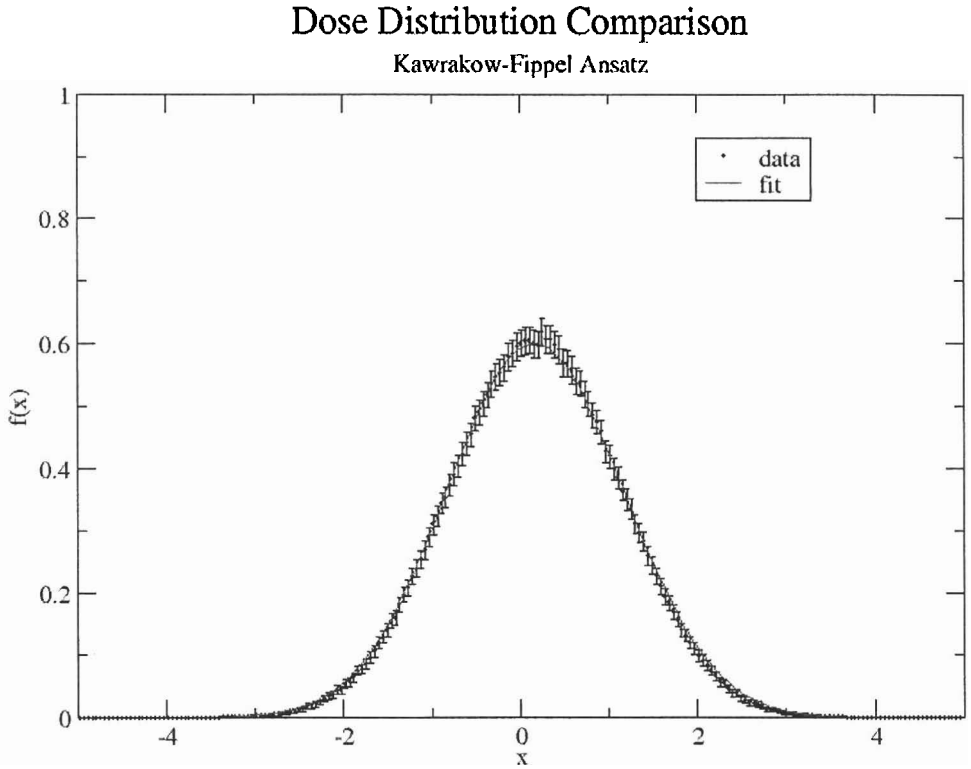

Figure 28: Fit of the dose distribution comparison. The fit determined that all of the voxels had a systematic bias of 0.17 standard deviations. In this case the standard deviation was equal to the combined statistical uncertainty, 0.7%. Therefore, the difference between the two distributions (0.1%) is insignificant. The reason the fit seems slightly too broad is due to the lack of accounting for cross-correlation when calculating the combined statistical uncertainty in each voxel.

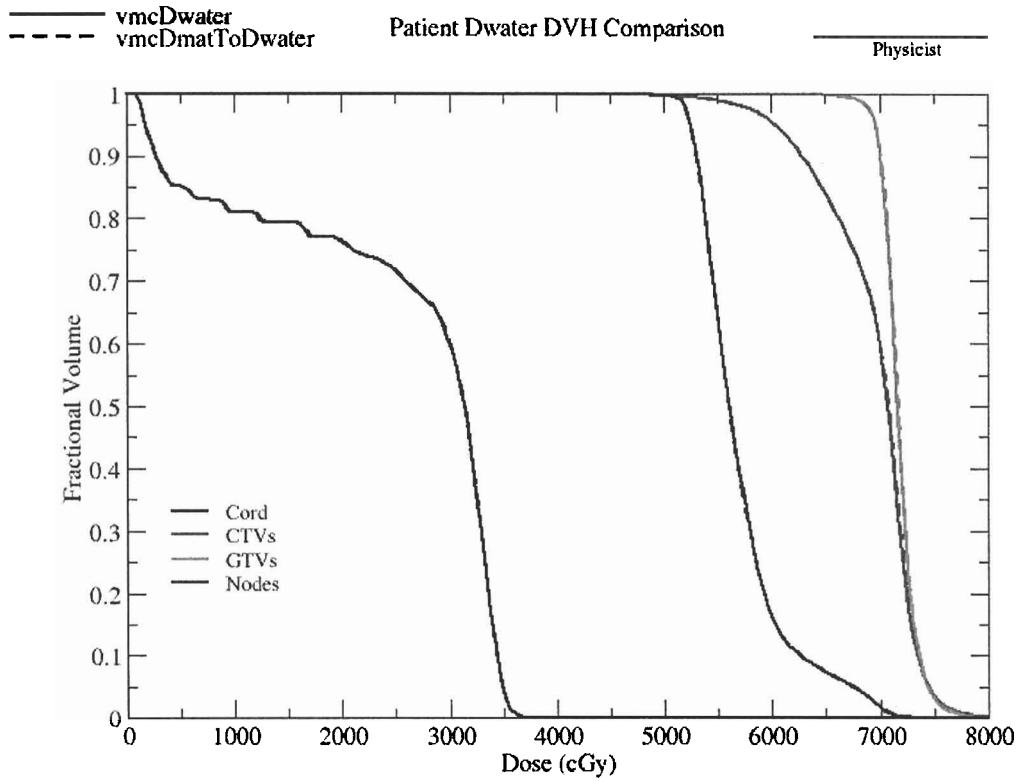

Figure 29: DVH comparing the VMC++ D_{water} dose calculation (solid lines) with the dose calculation produced by running the in-house conversion to D_{water} on the VMC++ $D_{material}$ dose calculation (dashed lines) for the spinal cord (black), the nodes (blue), the CTVs (red), and the GTVs (green). Differences between the two calculations are nearly indiscernible.

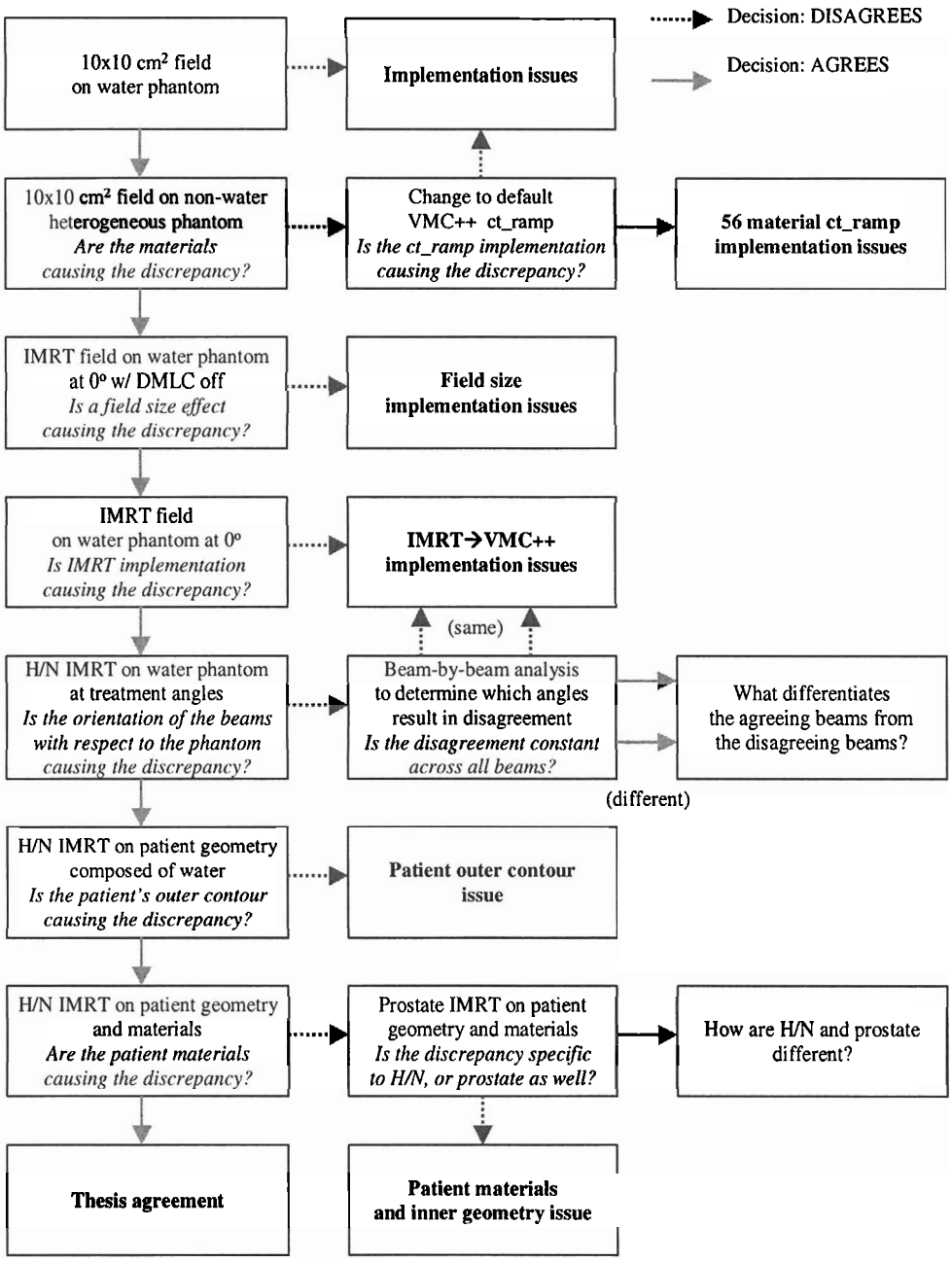

Figure 30: Flow chart to determine the cause of discrepancy between DOSXYZnrc and VMC++ for H/N patient case. Solid green arrows represent agreement between the two

codes, while dashed red arrows represent disagreement. Each box represents a test case between the two codes which has been designed to answer the respective question in italics.

Appendix

Sample VMC++ Input File

```
# Autogenerated config file
:start general source:
monitor units source_1 = 1
:start vcuPhaseSpace source:
my name = source_1
gantry angle = 270                          // set the angles
couch angle = 0.000000
collimator angle = 345
isocenter coordinates = 22.13 31.25 123.70 // set the isocenter
file name = unknownSourceFileName           // phase space file name
phaseSpaceZ = unknownSourceDistance         // distance to phase space
:stop vcuPhaseSpace source:                      plane; to be set later
:stop general source:
:start geometry:
:start XYZ geometry:
my name = xyz
method of input = MMC-PHANTOM
phantom file = unknownPhantomFileName       // phantom file name;
:stop XYZ geometry:                              to be set later
:stop geometry:
:start scoring options:
start in geometry: xyz
:start dose options:
score in geometries: xyz
score dose to water: no                     // Dwater or Dmat set here
:stop dose options:
:start output options xyz:
dump dose: 1
:stop output options xyz:
:stop scoring options:
:start MC Parameter:
automatic parameter = yes
:stop MC Parameter:
:start MC Control:
required uncertainty = 1.5                  // nominal uncertainty
insertNCaseHere
:stop MC Control:
:start variance reduction:
repeat history = 0.041
split photons = 1
photon split factor = unknownPhotonSplitFactor // set photon-split
:stop variance reduction:                            factor
:start quasi:
base = 2                                    // set quasi-random values
dimension = 60
skip = 1
:stop quasi:
```

List of References

[1] "The history of cancer," American Cancer Society, Inc. www.cancer.org/docroot/cri/content/cri_2_6x_the_history_of_cancer_72.asp?sitearea=cri (2002).

[2] T. J. Deeley and B. Hale, "The past seventy-five years of radiotherapy," Brit. J. Rad. **46**, 906-910 (1973).

[3] P. C. Goodman, "The new light: discovery and introduction of the X-ray," Radiology Centennial, Inc. www.xray.hmc.psu.edu/rci/centennial.html (1993).

[4] ICRU-29, "Dose specifications for reporting external beam therapy with photons and electrons," International Committee on Radiation Units and Measurement, Bethesda, Maryland (1978).

[5] J. Cunningham, "Quality assurance in dosimetry and treatment planning," Int. J. Radiat. Onc., Biol., Phys. **10**, 105-109 (1984).

[6] M. R. Arnfield, C. H. Siantar, J. Siebers, P. Garmon, L. Cox, and R. Mohan, "The impact of electron transport on the accuracy of computed dose", Med. Phys. **27**, 1266-1274 (2000).

[7] R. Mohan, "Why Monte Carlo?" in *XII International Conference on the Use of Computers in Radiotherapy*, Salt Lake City, Utah, Medical Physics Publishing, Madison, Wisconsin (1997).

[8] M. E. Kowalok, "Adjoint methods for external beam inverse treatment planning," PhD thesis, The University of Wisconsin-Madison, Department of Medical Physics, Madison, Wisconsin (2004).

[9] J. Sempau, A. Sanchez-Reyes, F. Salvat, H. Oulad ben Tahar, S. B. Jiang, and J. M. Fernandez-Varea, "Monte Carlo simulation of electron beams from an accelerator head using PENELOPE," Phys. Med. Biol. **46**, 1163-1186 (2001).

[10] J. F. Williamson, "Monte Carlo simulation of photon transport phenomena: sampling techniques," in *Monte Carlo Simulation in the Radiological Sciences*, edited by R. L. Morin, 53-101, CRC Press, Boca Raton, Florida (1988).

[11] M. J. Berger, "Monte Carlo calculation of the penetration and diffusion of fast charged particles," in *Methods in Computational Physics, Volume 1*, edited by B. Alder, S. Fernbach, and M. Rotenburg, 135-215, Academic Press, New York (1963).

[12] E. W. Larsen, "A theoretical derivation of the condensed history algorithm," Ann. Nucl. Energy **19**, 701-714 (1992).

[13] A. F. Bielajew and D. W. O. Rogers, "Variance reduction techniques," National Research Council of Canada Report PIRS-0396 (1993).

[14] A. F. Bielajew, H. Hirayama, W. R. Nelson, and D. W. O. Rogers, "History, overview and recent improvements of EGS4," National Research Council of Canada, Report PIRS-0436 (1994).

[15] R. L. Ford and W. R. Nelson, "The EGS code system – Version 3," Report SLAC-210, Stanford Linear Accelerator, Stanford, California (1978).

[16] W. R. Nelson, H. Hirayama, and D. W. O. Rogers, "The EGS4 code system," Report SLAC-265, Stanford Linear Accelerator, Stanford, California (1985).

[17] I. Kawrakow, "Accurate condensed history Monte Carlo simulation of electron transport. I. EGSnrc, the new EGS4 version," Med. Phys. **27**, 485-498 (2000).

[18] I. Kawrakow and D. W. O. Rogers, "The EGSnrc code system: Monte Carlo simulation of electron and photon transport," National Research Council of Canada, Report PIRS-701, (2000).

[19] B. R. B. Walters and D. W. O. Rogers, "DOSXYZnrc users manual," National Research Council of Canada, Report PIRS-794 (2002).

[20] I. Kawrakow, M. Fippel, and K. Friedrich, "3D electron dose calculation using a Voxel based Monte Carlo algorithm (VMC)," Med. Phys. **23**, 445-457 (1996).

[21] I. Kawrakow and M. Fippel, "VMC++, a MC algorithm optimized for electron and photon beam dose calculations for RTP," in *Proceedings of the 22nd Annual International Conference of the IEEE*, Engineering in Medicine and Biology Society (2000).

[22] A. F. Bielajew and D. W. O. Rogers, "PRESTA: the parameter reduced electron-step transport algorithm for electron Monte Carlo transport," Nucl. Instr. and Methods **B18**, 165-181 (1987).

[23] M. Fippel, "Fast Monte Carlo dose calculation for photon beams based on the VMC electron algorithm," Med. Phys. **26**, 1466-1475 (1999).

[24] I. Kawrakow and M. Fippel, "Investigation of variance reduction techniques for Monte Carlo photon dose calculation using XVMC," Phys. Med. Biol. **45**, 2163 - 2184 (2000).

[25] I. Lux and L. Koblinger, *Monte Carlo particle transport methods: Neutron and Photon Calculations*, p222, CRC Press, Boca Raton, Florida (1991).

[26] I. Kawrakow, "VMC++, electron and photon Monte Carlo calculations optimized for Radiation Treatment Planning," in *Advanced Monte Carlo for Radiation Physics, Particle Transport Simulation and Applications: Proceedings of the Monte Carlo 2000 Meeting Lisbon*, edited by A. Kling, F. Barao, M. Nakagawa, L. Távora, and P. Vaz, 229-236, Springer, Berlin (2001).

[27] J. V. Siebers, P. J. Keall, J. Kim, and R. Mohan, "Performance benchmarks of the MCV Monte Carlo System," in *Proceedings of the 13th ICCR*, edited by T. Bortfeld and W. Schlegel, 129-131, Springer, Berlin (2000).

[28] D. W. Rogers, et al., "BEAM: a Monte Carlo code to simulate radiotherapy treatment units," Med. Phys. **22**, 503-524 (1995).

[29] J. V. Siebers, et al., "Comparison of EGS4 and MCNP 4b Monte Carlo codes for generation of photon phase space distributions for a Varian 2100C," Phys. Med. Biol. **44**, 3009-3026 (1999).

[30] F. Haryanto, M. Fippel, W. Laub, O. Dohm, and F. Nusslin, "Investigation of photon beam output factors for conformal radiation therapy—Monte Carlo simulation and measurements," Phys. Med. Biol. **47**, N133-143 (2002).

[31] B. Fraass, K. Doppke, M. Hunt, G. Kutcher, G. Starkschall, R. Stern, and J. Van Dyke, "American Association of Physicists in Medicine Radiation Therapy Committee Task Group No. 53: Quality assurance for clinical radiotherapy treatment planning," Med. Phys. **25**, 1773-1829 (1998).

[32] D. A. Low, W. B. Harms, S. Mutic, and J. A. Purdy, "A technique for the quantitative evaluation of dose distributions," Med. Phys. **25**, 656-661 (1998).

[33] D. W. O. Rogers and R. Mohan, "Questions for comparisons of clinical Monte Carlo codes," in *Proceedings of the 13th ICCR*, edited by T. Bortfeld and W. Schlegel, 120-122, Springer, Berlin (2000).

[34] J. V. Siebers, P. J. Keall, A. E. Nahum, and R. Mohan, "Converting absorbed dose to medium to absorbed dose to water for Monte Carlo-based photon beam dose calculations," Phys. Med. Biol. **45**, 983-995 (2000).

[35] H. H. Liu and P. J. Keall, "Dm rather than Dw should be used in Monte Carlo treatment planning," Med. Phys. **29**, 922-924 (2002).

[36] ICRU-42, "Use of Computers in External Beam Radiotherapy Procedures with High-Energy Photons and Electrons," International Committee on Radiation Units and Measurements, Bethesda, Maryland (1987).